



USER GUIDE | PUBLIC

Document Version: 1912a – 2019-12-14

# SAP Leonardo IoT Services

# Content

<b>1</b>	<b>SAP Leonardo IoT</b>	<b>11</b>
1.1	About this API Reference	15
1.2	Application Names in Service URIs	18
1.3	Migrating from REST to OData	22
<b>2</b>	<b>Business Partner</b>	<b>25</b>
2.1	Organization	28
	Create an Organization	32
	Read an Organization	34
	Read all Organizations	36
	Read all Matching Organizations	39
	Update an Organization	41
	Delete an Organization	43
	Payload of CapabilityType "Organization"	44
	DeletedOrganization	48
2.2	Person	55
	Create a Person	59
	Read a Person	62
	Read all Persons	64
	Read all Matching Persons	67
	Update a Person	69
	Delete a Person	72
	Payload of CapabilityType "Person"	73
	DeletedPerson	77
2.3	Retrieving Enriched Person Data	85
2.4	BPRelationship	87
	Create a BPRelationship	89
	Read a BPRelationship	91
	Read all BPRelationships	92
	Delete a BPRelationship	95
	DeletedBPRelationship	96
2.5	BPRole	102
	Create a BPRole	104
	Read a BPRole	106
	Read all BPRoles	108
	Delete a BPRole	109
	DeletedBPRole	110

2.6	BUserRelationship. . . . .	116
	Create a BUserRelationship. . . . .	118
	Read a BUserRelationship. . . . .	120
	Read all BUserRelationships. . . . .	121
	Read all Matching BUserRelationships. . . . .	123
	Delete a BUserRelationship. . . . .	125
	DeletedBUserRelationship. . . . .	126
<b>3</b>	<b>Location. . . . .</b>	<b>133</b>
3.1	Location. . . . .	134
	Create a Location. . . . .	136
	Read a Location. . . . .	137
	Read all Locations. . . . .	138
	Read all Matching Locations. . . . .	140
	Update a Location. . . . .	142
	Delete a Location. . . . .	144
	DeletedLocation. . . . .	145
3.2	Geolocations Overview. . . . .	151
	Managing Geolocation Semantics. . . . .	151
	Locales. . . . .	152
	Geometry Types. . . . .	153
	Geolocation Spaces. . . . .	155
	Managing Geolocations. . . . .	164
	GeoTest. . . . .	181
	Space Hierarchies and Location Hierarchies. . . . .	190
	Authorization. . . . .	199
	Error Response Structures. . . . .	201
<b>4</b>	<b>Authorization. . . . .</b>	<b>202</b>
4.1	Object Instance-based Authorization. . . . .	205
	ObjectGroup. . . . .	206
	Root Object Group of a Tenant. . . . .	217
	CapabilityType. . . . .	221
	Capability. . . . .	225
	Filter Terms. . . . .	236
	ObjectGroupCollection. . . . .	241
	ObjectGroupCollectionItem. . . . .	249
	Capabilities. . . . .	260
4.2	Functional Authorization. . . . .	263
	Accessing SAP Leonardo IoT APIs Using OAuth Token. . . . .	269
4.3	Role Templates and Scopes. . . . .	271
<b>5</b>	<b>Onboarding Services. . . . .</b>	<b>281</b>

5.1	Tenant. . . . .	282
	Read a Tenant. . . . .	284
	Read All Tenants. . . . .	285
5.2	Onboarding Organizations. . . . .	287
	Create Organization Assignment. . . . .	290
	Read Organization Assignment. . . . .	292
	Delete Organization Assignment. . . . .	293
5.3	Onboarding Persons. . . . .	295
	Create Person Assignment. . . . .	298
	Read Person Assignment. . . . .	302
	Update Person Assignment. . . . .	304
	Delete Person Assignment. . . . .	307
5.4	UserGroup. . . . .	309
	Create a UserGroup. . . . .	311
	Read a UserGroup. . . . .	313
	Read all UserGroups. . . . .	314
	Update a UserGroup. . . . .	315
	Delete a UserGroup. . . . .	317
5.5	IdentityZone. . . . .	318
	Create an IdentityZone. . . . .	319
<b>6</b>	<b>Data Protection and Privacy. . . . .</b>	<b>322</b>
6.1	Delete User Data. . . . .	324
6.2	Export Rule Data. . . . .	324
<b>7</b>	<b>Configuration: Metadata Objects. . . . .</b>	<b>328</b>
7.1	Reference Property Set Type. . . . .	331
<b>8</b>	<b>Configuration: OData Service. . . . .</b>	<b>334</b>
8.1	Package Configuration. . . . .	337
	Read Metadata of Package. . . . .	338
	Create a Package. . . . .	341
	Read a Package. . . . .	343
	Read all Packages. . . . .	345
	Update a Package. . . . .	349
	Delete a Package. . . . .	350
8.2	Thing Configuration. . . . .	352
	Read Metadata of Thing Configuration. . . . .	352
	Thing Type. . . . .	368
	Thing Type Association. . . . .	394
	Thing Template. . . . .	406
	Property Set Type. . . . .	434



	Property Type. . . . .	476
	Annotation. . . . .	501
8.3	Event Configuration. . . . .	512
	Read Metadata of Event Configuration. . . . .	514
	Event Type. . . . .	520
	Event Property Set Type. . . . .	557
8.4	Thing Type and Sensor Type Mapping. . . . .	577
	Create a Thing Type and Sensor Type Mapping. . . . .	581
	Read a Thing Type with its Sensor Type Mappings. . . . .	585
	Read a Thing Type with its Sensor Types used in Mappings. . . . .	588
	Update Sensor Type Mappings or Sensor Types in the Thing Type. . . . .	593
	Delete a Thing Type with its Sensor Type Mappings. . . . .	596
<b>9</b>	<b>Custom Master Data. . . . .</b>	<b>598</b>
9.1	Object Types. . . . .	598
	Create an Object Type. . . . .	599
	Update an Object Type. . . . .	604
	Read an Object Type. . . . .	606
	Read all Object Types. . . . .	611
	Read all Object Types of a Package. . . . .	615
	Delete an Object Type. . . . .	620
	Standard Object Types. . . . .	621
9.2	Object Type Association. . . . .	622
	Create Object Type Association. . . . .	624
	Read all Object Type Associations. . . . .	627
	Delete Object Type Association. . . . .	630
9.3	Object Type Instances. . . . .	631
	Metadata of Object Type Instances. . . . .	632
	Create an Object Type Instance. . . . .	634
	Update an Object Type Instance. . . . .	635
	Read an Object Type Instance. . . . .	637
	Read all Object Type Instances. . . . .	638
	Delete an Object Type Instance. . . . .	642
	\$Batch. . . . .	643
9.4	Instance Associations. . . . .	646
	Create an Instance Association. . . . .	647
	Read an Instance Association. . . . .	649
	Read all Instance Associations. . . . .	650
	Delete an Instance Association. . . . .	652
	\$Batch. . . . .	653
	Create Instance Association with Standard Object Types. . . . .	656
9.5	Role Templates and Scopes. . . . .	657

9.6	Value Data Types for Object Types and Properties. . . . .	659
9.7	Limitations. . . . .	661
9.8	Audit Logs. . . . .	661
9.9	Data Protection and Privacy. . . . .	665
<b>10</b>	<b>Custom Master Data Composite Services. . . . .</b>	<b>667</b>
10.1	Composite Custom Master Data Basic Service. . . . .	667
	Read Metadata of Composite Custom Master Data Basic Service. . . . .	668
	Read List of Things. . . . .	670
10.2	Composite Custom Master Data Details Service. . . . .	675
	Read Metadata of Composite Custom Master Data Details Service. . . . .	675
	Read List of Things. . . . .	678
<b>11</b>	<b>Actions. . . . .</b>	<b>688</b>
11.1	Create an Action. . . . .	689
11.2	Update an Action. . . . .	691
11.3	Read an Action. . . . .	692
11.4	Read all Actions. . . . .	694
11.5	Delete an Action. . . . .	696
11.6	Delete Actions. . . . .	697
11.7	Create an Action Event. . . . .	698
11.8	Update an Action Event. . . . .	700
11.9	Get Pending/all Action Events. . . . .	701
<b>12</b>	<b>Thing. . . . .</b>	<b>705</b>
12.1	Thing Data. . . . .	705
12.2	Time Series Data Ingestion. . . . .	707
	Monitoring Time Series Data Ingestion Delay. . . . .	710
12.3	Onboarding Things . . . . .	715
	Create a Thing. . . . .	716
	Create Multiple Things. . . . .	727
	Read a Thing by Alternate ID. . . . .	729
	Read a Thing. . . . .	731
	Read Thing Configuration Details for a Thing. . . . .	734
	Update a Thing. . . . .	739
	Delete a Thing. . . . .	742
	Read all Things. . . . .	743
	Payload of CapabilityType "Thing". . . . .	746
12.4	Sensors and Things. . . . .	748
	Flexible Mapping (OData). . . . .	751
	Error Messages. . . . .	796
12.5	Grouping of Things. . . . .	801

	Thing Group. . . . .	802
	Thing Group Element. . . . .	819
	Thing Group Description. . . . .	822
12.6	Association of Things. . . . .	824
	Read Metadata of Thing Association. . . . .	826
	Create Thing Association. . . . .	829
	Read All Associations of a Thing. . . . .	831
	Read Details of a Thing Association. . . . .	834
	Delete Association of a Thing. . . . .	836
12.7	Read Version Information of Thing Service. . . . .	837
12.8	HTTP Status Codes for Thing Services. . . . .	838
<b>13</b>	<b>Thing: Master Data and Parameters Data. . . . .</b>	<b>840</b>
13.1	Read Metadata for a Thing. . . . .	840
13.2	Create Data for a Thing. . . . .	845
13.3	Create Data Using Alternate Thing ID. . . . .	848
13.4	Read Data for a Thing. . . . .	849
13.5	Delete Data for a Thing. . . . .	853
13.6	Read Property Set Data. . . . .	854
<b>14</b>	<b>Thing: Reference Property Data. . . . .</b>	<b>858</b>
14.1	Create Data for Reference Properties. . . . .	859
14.2	Read Data for Reference Properties. . . . .	862
14.3	Read Reference Properties Data for a List of Property Sets. . . . .	864
14.4	Delete Data for Reference Properties. . . . .	868
<b>15</b>	<b>Thing: Time Series Data. . . . .</b>	<b>870</b>
15.1	Time Series Store. . . . .	872
	Create Time Series Data Store for Tenant. . . . .	873
	Create Retention Period for Time Series Data Store. . . . .	874
	Read Retention Period of Time Series Data. . . . .	876
	Create Time Series Data or Derived Data for a Thing. . . . .	877
	Create Time Series Data or Derived Data Using Alternate Thing ID. . . . .	881
	Read Time Series Data or Derived Data for a Thing. . . . .	883
	Delete Time Series Data or Derived Data. . . . .	889
15.2	Time Series Aggregate Store. . . . .	890
	Read Time Series Data for a Property Type of a Thing Applying M4 Algorithm. . . . .	894
	Read Time Series Data for Multiple Property Types of a Thing Applying M4 Algorithm. . . . .	897
	Recalculate M4 Aggregates for Time Series Data. . . . .	900
	Read Snapshot Data of a Thing. . . . .	901
	Read Snapshot Data Within a Time Range for a Thing. . . . .	906
15.3	Time Series Cold Store. . . . .	910

	Create Time Series Data or Derived Data for a Thing. . . . .	911
	Read Time Series Data or Derived Data for a Thing. . . . .	913
	Delete Time Series Data or Derived Data. . . . .	915
<b>16</b>	<b>Thing Time Series Data: OData Service. . . . .</b>	<b>917</b>
16.1	Read Metadata. . . . .	918
16.2	Aggregates. . . . .	921
	Read Aggregates of Measurements. . . . .	922
16.3	Measurements. . . . .	931
	Read Single Measurement. . . . .	931
	Read Multiple Measurements. . . . .	933
<b>17</b>	<b>Cloud-to-Cloud Interoperability. . . . .</b>	<b>942</b>
17.1	Artifacts for Cloud-to-Cloud Interoperability. . . . .	942
	Create Artifacts. . . . .	943
<b>18</b>	<b>Thing Hierarchy. . . . .</b>	<b>945</b>
18.1	Thing Hierarchy. . . . .	946
	Thing Hierarchy. . . . .	947
	Thing Hierarchy Element. . . . .	973
	Thing Hierarchy Description. . . . .	979
18.2	Read Version Information of Thing Hierarchy Service. . . . .	980
18.3	HTTP Status Codes for Thing Hierarchy. . . . .	981
<b>19</b>	<b>Event Services. . . . .</b>	<b>982</b>
19.1	Standard or Predefined Events. . . . .	983
	Create an Event. . . . .	985
	Read an Event. . . . .	987
	Read all Events. . . . .	989
	Update an Event. . . . .	992
	Update Status for Multiple Events. . . . .	993
	Delete an Event. . . . .	994
	Delete all Events. . . . .	995
	Read Version Information of Event Service. . . . .	996
	HTTP Status Codes for Event Services. . . . .	997
19.2	User-defined Events based on OData Event Type. . . . .	998
	Read Metadata for Event of an Event Type. . . . .	999
	Create Events. . . . .	1001
	Read Events and Event Data. . . . .	1008
	Delete an Event. . . . .	1013
	Read Metadata for Event. . . . .	1014
	Read Events. . . . .	1021
	Read Event Correlations. . . . .	1025

	Read Events with a Correlation ID. . . . .	1028
	Delete an Event Using Correlation ID. . . . .	1031
<b>20</b>	<b>File. . . . .</b>	<b>1033</b>
20.1	Upload a File. . . . .	1034
20.2	Read a File. . . . .	1036
20.3	Read Metadata. . . . .	1038
20.4	Delete a File. . . . .	1039
20.5	Read Version Information of File Service. . . . .	1041
20.6	HTTP Status Codes for File Services. . . . .	1042
<b>21</b>	<b>Composites. . . . .</b>	<b>1043</b>
21.1	Composite Things: REST Service. . . . .	1043
	Read Thing Details with Property Set Type Data. . . . .	1044
	Read Version Information of Composite Service. . . . .	1046
	HTTP Status Codes for Composites. . . . .	1047
21.2	Composite Things: OData Service. . . . .	1047
	Read Metadata of Composite Things. . . . .	1049
	Read Details of a Property Set Type. . . . .	1052
	Read all Property Set Types. . . . .	1053
	Read all Property Sets for a Property Set Type. . . . .	1056
	Read Things for Property Set of a Property Set Type. . . . .	1058
21.3	Composite Events: OData Service. . . . .	1062
	Read Metadata of Composite Events. . . . .	1063
	Read Details of all Event Types. . . . .	1067
	Read Details of an Event Type . . . . .	1071
	Read Events and Event Data for Property Set of a Property Set Type. . . . .	1073
	Read Details of all Events for an Event Type. . . . .	1080
	Read Details of an Event for an Event Type . . . . .	1089
21.4	Composite Thing Configuration for a Specific Thing Hierarchy: OData Service. . . . .	1093
	Read Metadata. . . . .	1095
	Read all Hierarchies. . . . .	1098
	Read Details of a Thing Hierarchy. . . . .	1100
	Read Thing Hierarchy Elements. . . . .	1101
	Read Thing Hierarchy Elements and Property Set Types. . . . .	1105
	Read Thing Hierarchy Elements, Property Set Types, and Properties. . . . .	1109
21.5	Thing Detail. . . . .	1121
	Read Metadata of Thing Detail. . . . .	1122
	Read Details of a Thing. . . . .	1129
	Read Property Set Data for a Thing. . . . .	1132
21.6	Advanced Thing List. . . . .	1136
	Read Metadata of Thing List. . . . .	1137

	Read List of Things. . . . .	1148
	Read Property Set Data and Events for a Thing. . . . .	1162
21.7	Event List: OData Service. . . . .	1172
	Read Metadata of Event Type. . . . .	1173
	Read Events of Event Type for Customer. . . . .	1175
21.8	Events Aggregate: OData Service. . . . .	1181
	Read Metadata of Events Aggregate. . . . .	1182
	Read Aggregate of Events. . . . .	1183
<b>22</b>	<b>Query Modeler Dynamic OData services. . . . .</b>	<b>1187</b>
22.1	Services. . . . .	1188
	Retrieve the List of Available Services. . . . .	1188
	Retrieve the List of Entity Sets of the Service. . . . .	1189
	Retrieve the Metadata of the Service. . . . .	1191
	Retrieve Personal Data about Data Subjects. . . . .	1194
22.2	Things. . . . .	1196
	Retrieve a Thing. . . . .	1196
	Retrieve the List of Things. . . . .	1197
	Delete a Thing. . . . .	1199
22.3	Facts. . . . .	1200
	Retrieve a Fact. . . . .	1200
	Retrieve the List of Facts. . . . .	1201
	Delete a Fact. . . . .	1203
<b>23</b>	<b>Monitoring and Logging. . . . .</b>	<b>1205</b>
23.1	Read all Correlations. . . . .	1205
23.2	Read Status of a Correlation ID. . . . .	1208
23.3	Read Log Details of a Correlation. . . . .	1210
23.4	Read Details of a Log. . . . .	1212
<b>24</b>	<b>Filter Conditions (\$filter). . . . .</b>	<b>1214</b>
<b>25</b>	<b>Retrieving List of Records. . . . .</b>	<b>1220</b>
<b>26</b>	<b>Draft Mode for Service Requests. . . . .</b>	<b>1222</b>
<b>27</b>	<b>Reserved Keywords for Configuration and Thing Services. . . . .</b>	<b>1223</b>
<b>28</b>	<b>Value Data Types for Property Types and Properties. . . . .</b>	<b>1229</b>
<b>29</b>	<b>Value Helps. . . . .</b>	<b>1238</b>
29.1	ISO Language Codes. . . . .	1242
29.2	Country Codes. . . . .	1244
<b>30</b>	<b>HTTP Status Codes. . . . .</b>	<b>1255</b>



# 1 SAP Leonardo IoT

Introducing SAP Leonardo Internet of Things (IoT) services

## Introduction

The SAP Leonardo Internet of Things (IoT) services represent a network of physical objects known as things that are used to collect and exchange data. It helps to set up a thing model, companies, and service personnel in a company and their relationships. It also provides a collection of REST-based as well as OData-based services to store and retrieve data efficiently for a thing, company, and service personnel.

A thing is any physical object (for example a car or a robot - or any of its components or parts, regardless of size or complexity) that is relevant from a user or application perspective. Companies (also known as business partners) can manufacture and service a thing.

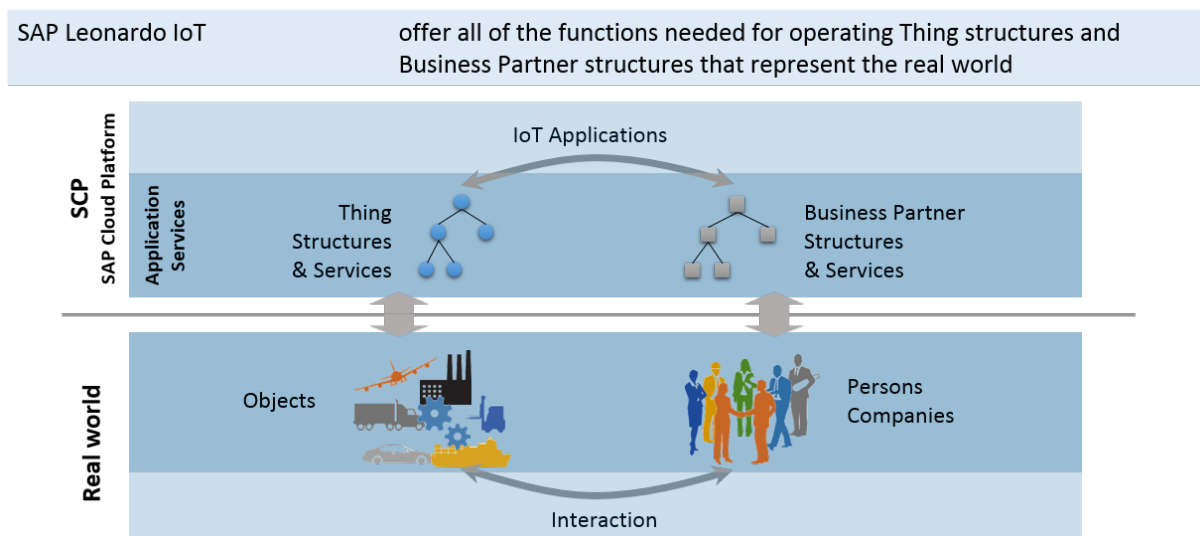
Here is a list of the most important groups of services available in the SAP Leonardo IoT space:

- Business Partner
- Location
- Authorization
- Configuration
- Thing
- Event
- File

Each of these groups comprises a number of services for detailed operations on the elements managed within the SAP Leonardo IoT Platform.

Of all the groups of services listed above, the Business Partner as well as the Thing services stand out. Together, they constitute a digital representation of both the real-world things (physical objects like machines or products of all kinds) as well as the persons and organizations that are interested in manufacturing, monitoring, controlling, or distributing these objects. Obviously, each kind of scenario dealing with the Internet of Things cannot exist without these two constituents, Business Partner and Thing. Therefore, SAP combines these essential entities in one software layer as the basis for reuse and as a unified and neutral cross-business

approach with respect to these central elements. The following picture illustrates the relations between business partners and things, the real world and its representation on SAP Cloud Platform:



As a prerequisite for the different types of access to things, a detailed modeling framework is required to define ownership and access rights for the objects involved as follows:

- **Ownership:**

One of the first and central questions in an IoT scenario is: What are the objects that I can access, and how can my objects be protected against unauthorized access by third parties (and how does the system ensure that I don't accidentally access objects outside of my own area of responsibility)?

To accomplish this, SAP Leonardo IoT introduces a tenant concept. A tenant serves as a container which hosts all the objects that belong to one particular cloud platform consumer (that is, an organization that has been onboarded on the SAP Leonardo IoT platform). Members of the onboarded organization can access the objects contained in the tenant to which they have been assigned, and they are restricted to objects in that single tenant. Objects in other tenants cannot be accessed by cloud platform consumers. This strict separation can only be overcome by system administrators who are operating the platform for the cloud platform provider.

- **Access rights:**

SAP Leonardo IoT platform comes with a two-dimensional authorization concept, object instance-based authorizations and functional authorizations:

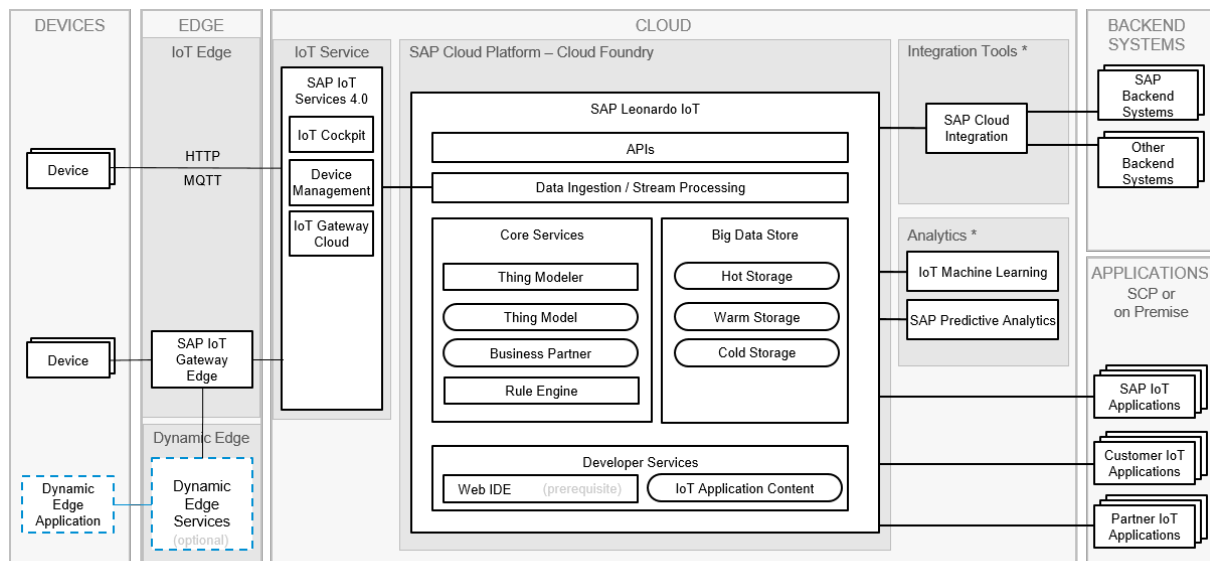
- With object instance-based authorizations (referred to as "capabilities"), you can set up relationships between members of your organization and the objects that each member may access. The maximum number of accessible objects is the number of objects that exist within the tenant to which your organization is assigned. As long as no such relationships have been established, a user would have no chance to see if there are any objects at all within the tenant.
- With functional authorizations (referred to as "scopes"), you can control the type of access for which a user is entitled with respect to the objects he or she may access. The following scopes are supported: read, write, delete. In addition, there is a fourth scope dealing with the physical deletion of objects that have already been logically deleted. This fourth scope, however, is exclusively granted to administrative personnel operating the platform.

## Architecture Overview

The following diagram shows the different layers and components of SAP Leonardo IoT and how they interact with other software layers as well as real-world devices in a sample system landscape.

### Note

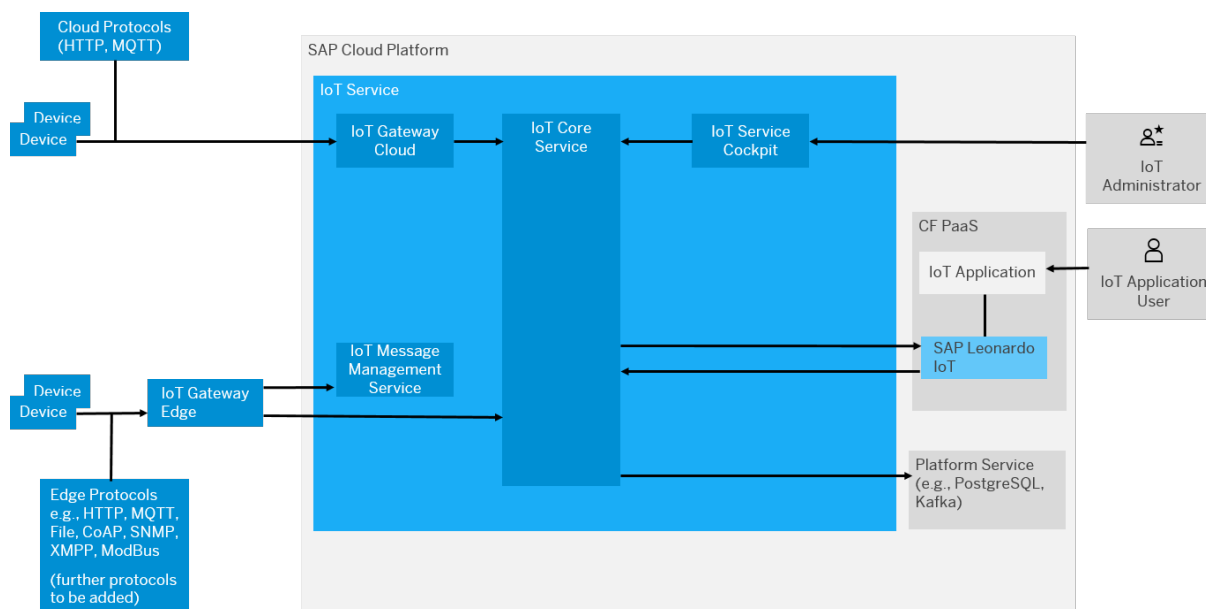
The diagram contains a small number of elements that are not yet available but may be part of a later release. These elements are marked with an asterisk (\*).



\* future option – not yet supported

SAP Leonardo IoT- Architecture and Landscape Overview

The following diagram shows where and how SAP Leonardo IoT is integrated into the overall architecture of SAP Cloud Platform Internet of Things for the Cloud Foundry Environment, which serves as the runtime environment for SAP Leonardo IoT:



SAP Cloud Platform Internet of Things - Architecture Overview

## Reuse Aspects

Although the Internet of Things can be seen as an opportunity for entering completely new areas of business activities, there is even more to it than just that. Even though the Internet of Things is for sure a new paradigm of generating and handling data, it does not replace the traditional world of business as we know it. Rather, the Internet of Things can be regarded as a complement, or enhancement, of already existing business scenarios and the IT infrastructure used for managing these scenarios. The enormous amount of data that is generated by the Internet of Things can be connected to existing infrastructures and applications. These applications can then take advantage of the massively increased data basis provided by IoT-based solutions and the insights that are made possible on this basis. Here are some examples for such enhancements of familiar business areas:

- **Logistics, Transportation Management:** With IoT-based solutions like SAP Networked Logistics Hub or SAP Connected Vehicles, you can access real-time data about the current traffic situation. With that, you can optimize the routing of the vehicles that you use to deliver goods to your customers.
- **Public Sector:** Local or regional authorities can use the same traffic data not only to avoid current traffic jams but to develop a mid-term or long-term strategy of an improved traffic situation, which would bring advantages for local companies as well as for the daily life of people.
- **Logistics, Warehousing, Manufacturing:** IoT-based solutions dealing with predictive maintenance give you the chance of early knowing when to produce, put on stock, or deliver certain goods to the place where they are needed. With that, you can avoid downtimes and the costs that are usually related to such downtimes. In addition, this lets you optimize service offerings related to your products that are in use at your customers' sites.
- **Agriculture:** Sensor data from the Internet of Things can help farmers to always find the best point in time when to start the seed of certain types of crop or vegetable as well as to find the optimal treatment for the soil on their fields.

## Related Information

[Business Partner \[page 25\]](#)

[Location \[page 133\]](#)

[Object Instance-based Authorization \[page 205\]](#)

[Configuration: OData Service \[page 334\]](#)

[Event Services \[page 982\]](#)

[File \[page 1033\]](#)

## 1.1 About this API Reference

Introduces the formal structure of this API reference and the naming conventions used

### Formal Structure of this API Reference

In this reference, we have grouped the content by the following scheme:

1. Topic introduction
2. Service
3. Service methods
4. Deleted Service (where applicable)
5. Deleted Service methods (where applicable)

#### ❖ Example

Here is an example for this scheme taken from the Business Partner section:

1. [Business Partner \[page 25\]](#)
2. [Person \[page 55\]](#)
3. [Create a Person \[page 59\]](#)
4. [DeletedPerson \[page 77\]](#)
5. [Delete a DeletedPerson \[page 84\]](#)

## Naming Conventions

The names of the service method chapters in this documentation follow the "CRUD" scheme that is common in the realm of REST services, that is, "create/read/update/delete". The service chapter names match the technical HTTP requests according to the following table:

Service Methods and HTTP Requests

Service Method	HTTP Request	Description
Create	<i>POST</i>	Creating a new instance of a database entity managed by a service
Read	<i>GET</i>	Retrieve a single instance of a database entity managed by a particular service
Read all	<i>GET</i>	Retrieve multiple database entities managed by a particular service
Update	<i>PUT</i>	Modify a database entity managed by a service
Delete	<i>DELETE</i>	Delete a database entity managed by a service (logically or physically, depending on the service)

## Case Sensitivity

Note that capital and non-capital letters in the technical names used by the services must be used exactly as given in this documentation. This affects payload parameters, request parameters, filter values and so on. In other words, the services that make up the SAP Leonardo IoT solution are case-sensitive. Programming against the API without taking care of the proper letter case leads either to HTTP errors or, which is even worse, to unpredictable results.

### ❖ Example

Here is an example of a collective GET request for retrieving a subset of persons stored in the system:

```
/Persons?$top=25&$orderby='personName.familyName' desc
```

In the following example, the system will complain about an unknown parameter because of a misspelled "T" instead of "t" for the \$top parameter:

```
/Persons?$Top=25&$orderby='personName.familyName' desc
```

In the following example, the system returns an HTTP error because the correct capitalization of the referenced payload attribute has not been observed:

```
/Persons?$top=25&$orderby='PersonName.FamilyName' desc
```

## Scope and Application Names

When it comes to the names of scopes, which are required as functional authorizations for accessing database entities in a particular way, the problem arises that these names consist of multiple components which are (at least partially) depending on the individual conditions that are in effect for a given deployment landscape. Therefore, we use a generic syntax that makes use of symbolic rather than technical names. The mapping between symbolic and technical scope names is explained in the [Functional Authorization \[page 263\]](#) section.



## Encoding Special Characters

There are certain specifications to be followed while using the service request URIs described in this document. Some special characters are not supported in the request URIs and using them will fetch unexpected results. Instead of using such special characters directly in the request URI, you must encode them as follows:

Special Characters	ASCII Code
{	%7B
}	%7D
\	%5C

For example, the following is an invalid request URI that will **not** fetch the expected result:

```
$filter=JSON1_adt%20eq%20'{"glossary\":{"value\":"1-4\"}}'
```

Instead, the special characters in the request must be encoded as follows:

```
$filter=JSON1_adt%20eq%20'%7B%5C"glossary%5C":%7B%5C"value%5C":%5C"1-4%5C"%7D%7D'
```

## Peak Load Limitation

With the services described in this API reference document, you can gain access to every feature and functionality that is part of SAP Leonardo IoT. Especially when accessing time series data generated by sensors of your managed things, this may result in gigantic data streams, which potentially could endanger the stability and service readiness of the platform. With SAP Leonardo IoT offered as a hosted solution, this potential danger might grow even more serious in case several customers hosted on the same machine could try to load excessive amounts of data simultaneously.

To avoid this kind of problems, SAP has decided to limit the maximum allowed number of data streams per customer to a certain amount that depends on the customer's license model. For more information, see SAP note [2608168](#).

## Disclaimer

This API reference document presents all the services that are offered to customers and partners for use in their own applications based on SAP Leonardo IoT. When exploring the platform, you may discover additional services that are **not** explained in this reference. These undocumented services are mostly providing features and functions that are needed by the IoT-related ready-to-use apps that are shipped by SAP as a part of the SAP Leonardo IoT.

For these undocumented services, SAP does not offer any support whatsoever, nor does SAP give any warranty with regard to the future availability of such services. Programming against the API of undocumented services is completely at your own risk. We therefore highly recommend **not** using any undocumented services in your applications.

## Related Information

[Application Names in Service URIs \[page 18\]](#)

## 1.2 Application Names in Service URIs

Important information on application naming conventions in SAP Leonardo IoT

In this API reference, we are presenting all service methods offered by SAP Leonardo IoT together with the application to which the service belongs. The application names, however, are to a certain extent subject to individual decisions made by the system administrators on customer site. The application name given for each service method therefore has to be translated to the situation that is in effect for a particular system landscape. The following section explains how this translation works.

### Scopes and Application Names

In this API reference documentation, the scopes that are required for executing a service are listed for each service method in the *Permissions* part of the method documentation. Since the scopes are defined by the various applications that make up the SAP Leonardo IoT, each scope name consists of the application and the activity that is controlled by the scope. A formal description of a fully qualified scope looks as follows:

```
<xs_appname>[.<additional_resource>].<action>
```

Here is an explanation of the different parts of this formal description:

- **<xs\_appname>**: This name is defined during deployment. Within a productive system landscape, this name does not change over time, but it may vary from landscape to landscape. For example, the name used in a development landscape may not be the same as in the test landscape, and both may differ from the name in the productive landscape.
- **<action>**: Self-explanatory. According to the CRUD pattern known from databases, the following actions are supported: Create, Read, Update, Delete.

#### i Note

The four supported activities are abbreviated to their initial letter in lowercase (that is, `c`, `r`, `u`, and `d`). Some complex operations in the Leonardo IoT platform require that a user performing the operation needs a high number of permissions, which have to be passed to the system all at once. The short abbreviations make sure that this does not result in a field length overflow.

- **<additional\_resource>**: Identifier of a particular resource within an application for which a scope is defined. For example, the `<thing>.event.r` scope is required for a user to read the events that have been logged for a thing, where `event` is a resource that belongs to the `thing` application.

#### i Note

In this API reference documentation, we present the application-specific part of the scopes in a simplified form that makes it easy to understand the application in question, but still needs to be adapted to the

particular `<xs_appname>` that is in effect in your system landscape. The following symbolic names are used for this:

#### Symbolic Application Names for Scopes

Application	Symbolic Name
authorization	<code>&lt;auth&gt;</code>
business-partner	<code>&lt;bp&gt;</code>
location	<code>&lt;loc&gt;</code>
tenant-administration	<code>&lt;tenant&gt;</code>
thing	<code>&lt;thing&gt;</code>

Keep in mind that the angle brackets are used here to indicate the symbolic character of the name of the application defining the scope. In a real usage scenario, the entire symbolic name (including the angle brackets) must be replaced by the technical application name that has been chosen for your system landscape.

## Application Names and Service URIs

In the previous section, we have explained the relation between application names and scope qualifiers. A similar mechanism is in place (although not relevant for authorization) with respect to the service endpoints that are given in the *Base URI* part of each service overview chapter. Here, the formal description of a fully qualified URI looks like this:

```
http://<application name>.<application domain>[:<port number>][/<path>]
```

Like in the *Scopes and Application Names* section above, the `<application name>` as well as the `<application domain>` part are subject to arbitrary decisions of system administrators and cannot be considered a stable and uniform URI part across all installations around the globe. The reliable part, however, is the fact that by default, the `<application name>` part is identical with the application names listed above.

### i Note

In certain scenarios, the service URIs become longer as there could be too many request parameters in the URL to fetch the desired result set in the response payload. The long service URI results in large header size and the execution of the service fails with an error status code 400. Header size is limited to 16k. To overcome the long service URI, you can break the request into multiple requests.

For example, to [Read Reference Properties Data for a List of Property Sets \[page 864\]](#), you use the request URL `https://appiot-mds.cfapps.eu10.hana.ondemand.com/Things('503B8711B5D0445688E7FF7ACFC2177A')/core.automobiles:car/ReferenceProperties/'wheel_rr,Pressure&wheel_rr,RotationSpeed&wheel_fl,Pressure&wheel_fl,RotationSpeed'?timerange2015-03-10T08:00:00.000Z-2018-08-09T12:00:00.000Z`. The request might fail when there are huge number of reference properties in the request URL. To avoid larger header size, you can split a single request and hence, reduce the number of properties per request.

Here are some examples of service URIs to access the services:

#### Service URIs

Application	Service URI
<a href="#">Advanced Thing List [page 1136]</a> : OData	<code>https://advancedlist-thing-sap.cfapps.eu10.hana.ondemand.com/CompositeThings</code>
Authorization: <a href="#">ObjectGroup [page 206]</a>	<code>https://authorization.cfapps.eu10.hana.ondemand.com/ObjectGroups</code>
Business Partner: <a href="#">Organization [page 28]</a>	<code>https://business-partner.cfapps.eu10.hana.ondemand.com/Organizations</code>
Business Partner: <a href="#">Person [page 55]</a>	<code>https://business-partner.cfapps.eu10.hana.ondemand.com/Persons</code>
<a href="#">Composite Events: OData Service [page 1062]</a> : OData	<code>https://composite-events-odata.cfapps.eu10.hana.ondemand.com/CompositeEvents</code>
<a href="#">Composite Things: REST Service [page 1043]</a> : REST	<code>https://composite-things.cfapps.eu10.hana.ondemand.com/Composite/Things/PropertySet</code>
<a href="#">Composite Things: OData Service [page 1047]</a> : OData	<code>https://composite-things-odata.cfapps.eu10.hana.ondemand.com/CompositeThings</code>
<a href="#">Composite Thing Configuration for a Specific Thing Hierarchy: OData Service [page 1093]</a> : OData	<code>https://cs-hierarchy-meta-sap.cfapps.eu10.hana.ondemand.com/CompositesThingHierarchy</code>
<a href="#">Event Configuration [page 512]</a> : OData	<code>https://events-sap.cfapps.eu10.hana.ondemand.com/ES</code>
<a href="#">Package Configuration [page 337]</a> : OData	<code>https://config-package-sap.cfapps.eu10.hana.ondemand.com/Package</code>
<a href="#">Read Thing Configuration Details for a Thing [page 734]</a>	<code>https://apiot-mds.cfapps.eu10.hana.ondemand.com/Configuration</code>
<a href="#">Event List: OData Service [page 1172]</a>	<code>https://bpanalytics-event-sap.cfapps.eu10.hana.ondemand.com/BPAnalytics/EventType/v1/&lt;event type name&gt;</code>
<a href="#">Standard or Predefined Events [page 983]</a> : REST	<code>https://apiot-mds.cfapps.eu10.hana.ondemand.com/Events</code>

Application	Service URI
<a href="#">Events Aggregate: OData Service [page 1181]</a>	https://events-aggregate-sap.cfapps.eu10.hana.ondemand.com/EventsAggregate/v1/Events?timeRange=<from time>-<to time>&divisions=<divisions>
<a href="#">File [page 1033]: REST</a>	https://apiot-fs.cfapps.eu10.hana.ondemand.com/Files
<a href="#">Credentials [page 753]</a>	https://tm-data-mapping.cfapps.eu10.hana.ondemand.com/v1/credentials
<a href="#">Mapping [page 761]</a>	https://tm-data-mapping.cfapps.eu10.hana.ondemand.com/v1/mappings
<a href="#">Assignment [page 785]</a>	https://tm-data-mapping.cfapps.eu10.hana.ondemand.com/v1/assignments
<a href="#">Custom Master Data [page 598]</a>	https://config-customdata-sap.cfapps.eu10.hana.ondemand.com/CustomDataConfiguration
<a href="#">Custom Master Data Composite Services [page 667]</a>	https://composite-custom-master-data-sap.cfapps.eu10.hana.ondemand.com/CompositeMasterData/v1/Thingshttps://composite-custom-master-data-sap.cfapps.eu10.hana.ondemand.com/CompositeMasterData/Details/v1/Things
<a href="#">Actions [page 688]</a>	https://sap-iot-noah-live-action-modeler.cfapps.eu10.hana.ondemand.com/envPing
<a href="#">Thing Detail [page 1121]: OData</a>	https://details-thing-sap.cfapps.eu10.hana.ondemand.com/CompositeThings/v1/<thing type name>
<a href="#">Thing Hierarchy [page 945]: OData</a>	https://apiot-thing-hierarchy.cfapps.eu10.hana.ondemand.com/ThingHierarchy
<a href="#">Thing Configuration [page 352]: OData</a>	https://config-thing-sap.cfapps.eu10.hana.ondemand.com/ThingConfiguration
<a href="#">Onboarding Things [page 715]: REST</a>	https://apiot-mds.cfapps.eu10.hana.ondemand.com/Things
<a href="#">Thing Time Series Data: OData Service [page 917]</a>	https://analytics-thing-sap.cfapps.eu10.hana.ondemand.com/<property set type name>

Application	Service URI
<a href="#">Time Series Aggregate Store [page 890]</a> : REST	<code>https://appiot-mds.cfapps.eu10.hana.ondemand.com/Things</code>
<a href="#">Time Series Cold Store [page 910]</a> : REST	<code>https://appiot-coldstore.cfapps.eu10.hana.ondemand.com/Things</code>
<a href="#">Time Series Store [page 872]</a> : REST	<code>https://appiot-mds.cfapps.eu10.hana.ondemand.com/Things</code>
<a href="#">Value Helps [page 1238]</a>	<code>https://business-partner.cfapps.eu10.hana.ondemand.com/&lt;Value Help&gt;</code>

## Related Information

[Functional Authorization \[page 263\]](#)

## 1.3 Migrating from REST to OData

This topic is relevant only for users who have been using the REST-based Thing Model Configuration APIs provided as part of the predecessor product SAP IoT Application Enablement.

As of March 2019, the REST-based Thing Model configuration APIs are deprecated. We highly recommend using the new OData version of Thing Modeler configuration APIs, as this is the only version that will see future enhancements and new features.

### Deprecation End Date

Please note that the deprecation phase of REST-based Thing Model configuration ends by March 2020. After that date, these APIs will be decommissioned, and you cannot use them in your production environment. SAP will not provide any support services for these APIs after March 2020.



## Deprecated and Alternate APIs

	Deprecated APIs	Alternate APIs
Package Configuration APIs	<p><code>https://appcore-conf.cfapps.eu10.hana.ondemand.com/Configuration</code></p> <p><code>https://appcore-conf.cfapps.eu10.hana.ondemand.com/Configuration('&lt;package name&gt;')</code></p>	<p>CRUD APIs from <code>https://config-package-sap.cfapps.eu10.hana.ondemand.com</code></p>
Thing Configuration APIs	<p><code>https://apiot-mds.cfapps.eu10.hana.ondemand.com/Configuration/ThingTypes</code></p> <p><code>https://apiot-mds.cfapps.eu10.hana.ondemand.com/Configuration/ThingTypes('&lt;thingType name&gt;')</code></p> <p><code>https://apiot-mds.cfapps.eu10.hana.ondemand.com/Configuration/PropertySetTypes</code></p> <p><code>https://apiot-mds.cfapps.eu10.hana.ondemand.com/Configuration/PropertySetTypes('&lt;propertySetType name&gt;')</code></p> <p><code>https://apiot-mds.cfapps.eu10.hana.ondemand.com/Configuration/ThingTypes('&lt;thingType name&gt;')/PropertySetTypes</code></p> <p><code>https://apiot-mds.cfapps.eu10.hana.ondemand.com/Configuration('&lt;package name&gt;')/Annotations</code></p>	<p>CRUD APIs from <code>https://config-thing-sap.cfapps.eu10.hana.ondemand.com</code></p>

	Deprecated APIs	Alternate APIs
Event Configuration and events APIs	All APIs from <code>https://apiot-tes.cfapps.eu10.hana.ondemand.com</code>	CRUD APIs from <code>https://iot-events.cfapps.eu10.hana.ondemand.com</code>

## 2 Business Partner

A business partner is an entity with a particular role as an actor in a business scenario.

### Overview

Business partners can be defined on different levels of abstraction, and they can refer to either human beings or to a legal body. Here are some examples of different types of business partners:

- Company
- Employee
- Customer
- Supplier

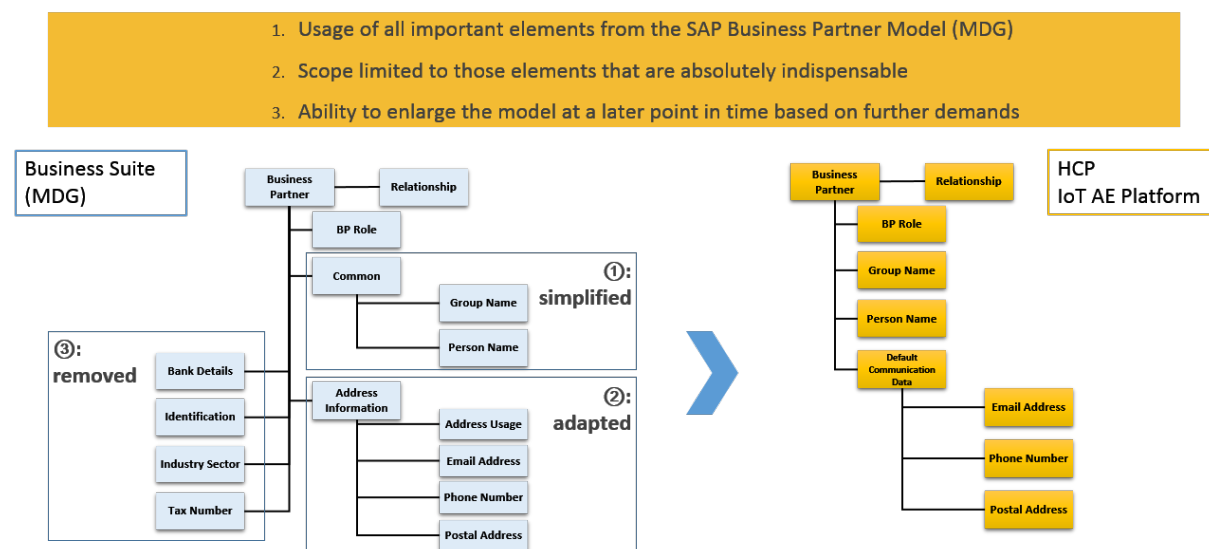
Business partners can have relationships both within and across the boundaries of a particular business partner type. For example, the concept of an employee only makes sense in combination with a company to which an employee belongs.

### Business Partner Concepts in SAP Master Data Governance and SAP Leonardo IoT

The concept of the business partner as a central entity that is needed for all kinds of business scenarios has a long tradition in SAP solutions. Therefore, it is desirable that different solutions follow an aligned approach with respect to the internal structure and the basic functions and features that are supported by each implementation of the business partner concept. To ensure seamless integration with existing solutions, the business partner concept of SAP Leonardo IoT is based on the well-established approach that is provided by the cross-application solution SAP Master Data Governance (MDG). The following picture illustrates the relationship between the business partner implementation both of SAP Master Data Governance and SAP Leonardo IoT. The central message here is that the SAP Leonardo IoT implementation is structurally aligned

with the SAP Master Data Governance solution while leaving out some of the more specific details which, however, might be added with a later release if needed:

## Reusing Important Elements of the MDG Business Partner Model



When comparing these two business partner concepts with each other, you may find the following aspects worth knowing:

- **Categories**  
SAP Master Data Governance supports three different business partner categories: Organization, Person, and Group. Of these three categories, SAP Leonardo IoT currently only supports the first two, Organization and Person.
- **Roles**  
Of the numerous business partner roles offered by SAP Master Data Governance and especially by the many solutions that make use of the business partner concept, SAP Leonardo IoT uses only two roles to further specify business partners that fall into the person category: Employee and Contact Person.
- **Relationships**  
As the word "partner" in the term "business partner" already implies, speaking of business partners makes sense only if there are two or more of them appearing in a particular business scenario. And with a multitude of business partners involved in a scenario, the question arises how the different business partners relate to each other. The answer to this question are the relationships that can be established to connect one business partner to another. Here, almost all combinations are possible:
  - 1:1  
Example: One employee acts as an assistant to his or her manager.
  - 1:N  
Example: One company has many employees.
  - M:N  
Example: Many companies are suppliers of many companies acting as customers.
  - No relationship  
Although the network of relationships is essential for the interactions between business partners, it is of course also possible that there are certain business partners within a scenario that don't have any direct relationship between each other. Example: Employee A works for company C, and employee B works for D, and C is customer of D, but A and B are not related.

While SAP Leonardo IoT reimplements the concepts of categories, roles, and relationships in an almost identical way as it is known from SAP Master Data Governance, there are also a few extensions which are specific for the SAP Leonardo IoT approach:

- **Persons and Users**  
In SAP Master Data Governance, business partners of type person always represent a person acting in a certain business scenario. In SAP Leonardo IoT, however, a person can also be associated with a user account that is needed for the person to logon to the system and perform the tasks for which system access is a prerequisite. This additional aspect is technically implemented by introducing a new type of relationship that is not known in SAP Master Data Governance, namely a relationship between a business partner and a user. Here, the user is an entity provided by a self-contained external identity provider solution like SAP Cloud Identity (SCI). For more information, see [BPUserRelationship \[page 116\]](#).
- **Organizations and Tenants**  
Similar to the technical extension of a person by a user, the business-oriented concept of an organization (whose instances typically represent a company or a department within a company) is also extended by a technical entity that goes beyond the concept known from SAP Master Data Governance. Here, the organization can be associated with a tenant. A tenant is an entity that belongs to the area of authorizations in SAP Leonardo IoT. A tenant serves as a virtual container for all the objects ("things") that belong to an organization. It protects the objects against all kinds of unauthorized access from outside, and at the same time ensures that a company can only access its own objects. For more information, see [Tenant \[page 282\]](#).

## Business Partner in SAP Leonardo IoT

In the realm of an application services infrastructure for the Internet of Things (IoT), a number of specific roles of business partners come into play, all of them belonging to the business partner as a company or an organization:

- **Cloud Platform Host**  
An organization providing the technical IT infrastructure (data centers including servers, storage solutions, high performance internet backbone connections, and so on) that is needed to offer a platform for cloud-based solutions.
- **Cloud Platform Provider (technical ID: T0)**  
An organization providing a technical platform offering cloud-based solutions for productive business support (for example, machinery administration in production plants of different companies around the globe.)
- **Cloud Platform Consumer (technical ID: T1)**  
An organization consuming the services of a cloud platform provider (for example, a manufacturer of industrial robots that are used in car production lines or smartphone assembly lines).
- **Cloud Platform Consumer 2nd Level (technical ID: T2)**  
An organization acting as a customer of a cloud platform consumer (for example, a car manufacturer or a smartphone manufacturer that have the industrial robots of the cloud platform consumer installed in their production lines).

## Example

ABC Systems, a technology company with expertise both in mechanical engineering and information technology, plans to offer a cloud-based software solution that enables providers of machine tools like Miller, Inc., to administer all the machines they have sold to customers around the world. One of these customers is CCC Ltd., a car manufacturer. ABC Systems, however, is only interested to offer the software solution as a service, but refuses to take care of the physical IT infrastructure needed to run the solution. Therefore, ABC Systems establishes a cooperation with Cloud IT, a company specialized in complex data center scenarios. It is then up to Cloud IT to handle the basic technical requirements such as short response times, 24/7 availability, load balancing, backup services, and so on.

In this example, the companies involved take over the following roles:

- Cloud Platform Host: Cloud IT
- Cloud Platform Provider: ABC Systems
- Cloud Platform Consumer: Miller Inc.
- Cloud Platform Consumer 2nd Level: CCC Ltd.

## Related Information

[Organization \[page 28\]](#)

[Person \[page 55\]](#)

## 2.1 Organization

An Organization represents a business partner of type organization that is assigned to one tenant.

Organizations are used to model the different companies or legal bodies that are involved in setting up an IoT services infrastructure. An Organization falls into one of the following categories:

- Cloud Platform Host
- Cloud Platform Provider
- Cloud Platform Consumer
- Cloud Platform Consumer 2nd Level

To identify an organization, both the `organizationName1` and the `emailAddress` attributes are required. The values for both fields shall be unique within the scope of the tenant to which the organization belongs.

By default, creating, accessing, or deleting an organization is only possible for organizations that are assigned to the same tenant to which the current user is logged on.

### i Note

The Organization services described here are focused on the isolated entity of a single organization. This is fine for accessing or modifying certain fields and attributes of an organization. However, for onboarding an organization (that is, creating a new organization and setting up relationships to associated objects), we recommend using the services described in the [Onboarding Organizations \[page 287\]](#) section.



## Prerequisites

The tenant is an attribute of the organization that must be provided upon creation of an organization. That is, to create an organization with a reference to a given tenant, the tenant name must be valid and the tenant must already exist in the system.

### Base URI:

- Formal description: `https://<server address>[:<port number>][<path>]/Organizations`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/Organizations`

## Methods

HTTP Method	Action	URI	Scopes
<i>POST</i>	<a href="#">Create an Organization [page 32]</a>	<code>/Organizations</code>	<code>&lt;bp&gt;.c</code> <code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.tenant.r</code>
<i>GET</i>	<a href="#">Read an Organization [page 34]</a>	<code>/Organizations('&lt;ID&gt;')</code>	<code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.tenant.r</code>
<i>GET</i>	<a href="#">Read all Organizations [page 36]</a>	<code>/Organizations</code>	<code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.tenant.r</code>
<i>GET</i>	<a href="#">Read all Matching Organizations [page 39]</a>	<code>/Organizations</code>	<code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.tenant.r</code>
<i>PUT</i>	<a href="#">Update an Organization [page 41]</a>	<code>/Organizations('&lt;ID&gt;')</code>	<code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.u</code> <code>&lt;bp&gt;.tenant.r</code>
<i>DELETE</i>	<a href="#">Delete an Organization [page 43]</a>	<code>/Organizations('&lt;ID&gt;')</code>	<code>&lt;bp&gt;.d</code> <code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.tenant.r</code>

## Payload

Format: *JSON*

For PUT, POST, and GET requests, only the media type JSON is supported. The JSON for these methods has the following structure:

```
{ "basicData"      : {
  "businessPartnerID" : { "type": "string", "maxLength": 32 },
  "tenant"            : { "type": "string", "maxLength": 36, "required":
true },
  "etag"              : { "type": "string" }
},
  "organizationName" : {
  "organizationName1" : { "type": "string", "maxLength": 255, "required":
true },
  "organizationName2" : { "type": "string", "maxLength": 255 }
},
  "communicationData" : {
  "emailAddress"      : { "type": "string", "maxLength": 255, "required":
true },
  "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
  "mobilePhoneNumber"  : { "type": "string", "maxLength": 127 },
  "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
  "landlinePhoneNumber" : { "type": "string", "maxLength": 127 },
  "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
  "streetName"         : { "type": "string", "maxLength": 255 },
  "houseNumber"        : { "type": "string", "maxLength": 10 },
  "cityName"           : { "type": "string", "maxLength": 255 },
  "postalCode"         : { "type": "string", "maxLength": 10 },
  "country"            : { "type": "string", "maxLength": 2 },
  "countryDescription"  : { "type": "string", "maxLength": 60, "readOnly" :
true },
  "region"             : { "type": "string", "maxLength": 3 },
  "regionDescription"  : { "type": "string", "maxLength": 60, "readOnly" :
true },
  "longitude"          : { "type": "number" },
  "latitude"           : { "type": "number" }
},
  "objectGroup"       : { "type": "string", "maxLength": 32, "required":
true },
  "externalIDs"        : {
    value: [{ "type": "string", "maxLength": 60,
"required": "false" }]
  }
}
```

## i Note


A special mechanism has been implemented for the fields dealing with phone numbers in the communication data block of the payload with respect to the preceding numeric country code. Unlike manual dialing with a phone, the country code is **not** given as the standardized number for a country (for example, 1 for USA, 49 for Germany). Instead, the system expects the alphabetical ISO country code as a prefix (for example, "US" for USA, "DE" for Germany). The following fields are affected:

- mobileDestinationLocation
- landlineDestinationLocation

This mechanism is in effect for the service request types POST, GET, and PUT. You may look up the required ISO country codes in the [Country Codes \[page 1244\]](#) section of this documentation.

## i Note

For the `emailAddress` field, the following requirements apply:

- The content of the field must be a valid and well-formed e-mail address according to the rules set forth in [IETF RFC 2822](#) .
- The content of the field is validated internally by SAP Leonardo IoT with the help of the e-mail validator provided by the Hibernate framework.

As a consequence, the following applies: Although the technical field length is defined as 255 characters, neither the mailbox part (before the @ sign) nor the domain name (after the @ sign, but before the dot ".") may exceed a maximum length of 64 characters.

For more information, see the [Hibernate documentation](#) .

## External ID

In the payload, there is a special field that needs some additional explanation: `externalID`. This field is intended to hold an identifying string other than the purely technical `businessPartnerID` field, which the system needs for reasons of database consistency. In contrast to that, the `externalID` field can be used for providing an additional semantic identifier that users may be accustomed to from their daily business experience. For an organization, this field could be used to hold the number under which a company has been registered in the commercial register of the responsible authority. Or, if a company is listed at a stock exchange, you might want to enter the ticker symbol into that field.

## i Note

The following limitations apply:

- Although the `externalIDs` field has been implemented as an array, the system currently supports the assignment of no more than one external ID (or none).
- On database level, the `externalIDs` field is managed as a common attribute of both persons and organizations. Therefore, the values entered for this field must be unique not only within a tenant, but also across persons and organizations. In other words, it is not allowed to use a value for the `externalIDs` field of a person that has already been used for an organization and vice versa.

## Access Authorization for Organizations

Objects of type `Organization` are subject to the instance-based authorization concept that is implemented in SAP Leonardo IoT. For more information, see [Object Instance-based Authorization \[page 205\]](#). To grant access to an organization, a capability has to be derived from the predefined `CapabilityType` object for organizations. For more information on the `CapabilityType` payload needed for this purpose, see [Payload of CapabilityType "Organization" \[page 44\]](#).

## Related Information

[Onboarding Organizations \[page 287\]](#)

[Business Partner \[page 25\]](#)

## 2.1.1 Create an Organization

With this method, you create a new business partner of type organization. The system automatically assigns a unique ID to the new organization.

To access, create change and delete organizations you need to have the respective authorization. This instance authorization is realized through hierarchical ObjectGroups and ObjectGroupCapabilities for the object type `com.sap.appcore.organization`.

Also, you may decide that the new organization is assigned to a tenant different to the one where you are currently logged on. To accomplish this, you need to explicitly specify that tenant during creation of the organization. Note, however, that such a cross-tenant operation is possible only for top-level organizations, that is, an organization that acts as a cloud platform provider.

If the organization is successfully created, the server sends HTTP response code 201. Any other code indicates an error.

### Request

**URI:** `/Organizations`

**HTTP Method:** `POST`

**Permissions:**

`<bp>.c`

`<bp>.r`

`<bp>.tenant.r`

#### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross site request forgery token

#### Request Fields

Name	Required	Type	Description	Send As
organizationName	Yes	string	Semantic name of the organization	JSON
emailAddress	Yes	string	An email address used to get in contact with the organization.	JSON
tenant	No	string	If omitted, tenant of currently logged on user is used.	JSON

#### Request Example

```
/Organizations
```

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method has the following structure:

```
{ "basicData"      : {
  "businessPartnerID" : { "type": "string", "maxLength": 32 },
  "tenant"            : { "type": "string", "maxLength": 36, "required":
true },
  "etag"              : { "type": "string" },
  },
  "organizationName" : {
  "organizationName1" : { "type": "string", "maxLength": 255, "required":
true },
  "organizationName2" : { "type": "string", "maxLength": 255 }
  },
  "communicationData" : {
  "emailAddress"      : { "type": "string", "maxLength": 255, "required":
true },
  "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
  "mobilePhoneNumber"   : { "type": "string", "maxLength": 127 },
  "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
  "landlinePhoneNumber" : { "type": "string", "maxLength": 127 },
  "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
  "streetName"          : { "type": "string", "maxLength": 255 },
  "houseNumber"         : { "type": "string", "maxLength": 10 },
  "cityName"            : { "type": "string", "maxLength": 255 },
  "postalCode"          : { "type": "string", "maxLength": 10 },
  "country"             : { "type": "string", "maxLength": 2 },
  "countryDescription"  : { "type": "string", "maxLength": 60, "readOnly" :
true },
  "region"              : { "type": "string", "maxLength": 3 },
  "regionDescription"   : { "type": "string", "maxLength": 60, "readOnly" :
true },
  "longitude"           : { "type": "number" },
  "latitude"            : { "type": "number" }
  },
  "objectGroup"        : { "type": "string", "maxLength": 32, "required":
true },
  "externalIDs"        : {
    value: [{ "type": "string", "maxLength": 60,
"required": "false" }]
  }
}
```

### Note

For the `emailAddress` field, the following requirements apply:

- The content of the field must be a valid and well-formed e-mail address according to the rules set forth in [IETF RFC 2822](#).
- The content of the field is validated internally by SAP Leonardo IoT with the help of the e-mail validator provided by the Hibernate framework.

As a consequence, the following applies: Although the technical field length is defined as 255 characters, neither the mailbox part (before the @ sign) nor the domain name (after the @ sign, but before the dot ".") may exceed a maximum length of 64 characters.

For more information, see the [Hibernate documentation](#).

## Response

Format: [JSON](#)

### Response Header Fields

Name	Type	Description
Location	string	Path to the newly created Organization

### Response Status and Error Codes

Code	Reason
201	Organization created successfully

## Related Information

[Organization \[page 28\]](#)

## 2.1.2 Read an Organization

Retrieves a specified organization from the database

With this method, you send a request to the server to retrieve a particular organization specified by its ID.

If the organization is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

## Request

URI: `/Organizations('<businessPartnerID>')`

HTTP Method: [GET](#)

Permissions:

`<bp>.r`

`<bp>.tenant.r`

Request Parameters

Name	Required	Type	Description	Send As
businessPartnerID	Yes	string	ID of the organization	Path parameter

### Request Example

```
/Organizations('2c08c0b71187498498b113bf340186f8')
```

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Reason
200	Organization retrieved successfully

## Payload

Format: *JSON*

Only media type JSON is supported. The JSON for this method has the following structure:

```
{ "basicData" : {
  "businessPartnerID" : { "type": "string", "maxLength": 32 },
  "tenant" : { "type": "string", "maxLength": 36, "required":
true },
  "etag" : { "type": "string" },
  "organizationName" : {
    "organizationName1" : { "type": "string", "maxLength": 255, "required":
true },
    "organizationName2" : { "type": "string", "maxLength": 255 }
  },
  "communicationData" : {
    "emailAddress" : { "type": "string", "maxLength": 255, "required":
true },
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
    "mobilePhoneNumber" : { "type": "string", "maxLength": 127 },
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
    "landlinePhoneNumber" : { "type": "string", "maxLength": 127 },
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
    "streetName" : { "type": "string", "maxLength": 255 },
    "houseNumber" : { "type": "string", "maxLength": 10 },
    "cityName" : { "type": "string", "maxLength": 255 },
    "postalCode" : { "type": "string", "maxLength": 10 },
    "country" : { "type": "string", "maxLength": 2 },
    "countryDescription" : { "type": "string", "maxLength": 60, "readOnly" :
true },
    "region" : { "type": "string", "maxLength": 3 },
    "regionDescription" : { "type": "string", "maxLength": 60, "readOnly" :
true },
    "longitude" : { "type": "number" },
```

```

    "latitude"          : { "type": "number" }
  },
  "objectGroup"        : { "type": "string", "maxLength": 32, "required":
"true" },
  "externalIDs"        : {
    value: [{ "type": "string", "maxLength": 60,
"required": "false"}]
  }
}

```

## Related Information

[Organization \[page 28\]](#)

### 2.1.3 Read all Organizations

Retrieves a list of organizations from the database

With this method, you send a request to the server to retrieve a subset of all organizations, according to the filter criteria provided. For a collective request for a set of organizations, you have these options:

- You can restrict the set of organizations to be retrieved by filter criteria based on the organization name.
- You can define that the set of matching organizations shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

If the list of organizations is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

#### Note

Retrieving high numbers of objects from the database with only one collective GET request can lead to high system load and a significant increase of response times. Therefore, we highly recommend limiting the maximum number of returned objects to no more than 500 per method call. This can easily be accomplished with the help of the `$top` and `$skip` query parameters.

To avoid unwanted performance decline, collective GET requests are only supported up to a maximum of 500 objects that can be retrieved with one single call. Therefore, in use cases where it is crucial for you to ensure that all existing objects have been retrieved, you need to verify this by setting up a loop construct in your source code to repeat the request until the service method returns no further objects.

## Request

**URI:** `/Organizations`

**HTTP Method:** `GET`

**Permissions:**



<bp>.r

<bp>.tenant.r

## Query String Parameters

Name	Required	Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set
\$orderby	No	string	Field name to be used for sorting the result set in ascending or descending order
\$filter	No	string	Filter condition to be applied; valid fields are basicData/tenant, organizationName/organizationName1, organizationName/organizationName2, objectGroup, and roleCategory
\$count	No	integer	Returns the number of entries in the database, with filter conditions taken into account (if applicable)

Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
basicData/tenant, organizationName/organizationName1, organizationName/organizationName2, objectGroup, roleCategory	eq, ne

## Request Example

```
/Organizations
```

Retrieves all organizations that are available in the database. Note that this formally unrestricted request is still implicitly restricted to Organization objects in the current tenant for which the logged on user has an instance-based access authorization.

```
/Organizations?$top=25&$orderby='organizationName.organizationName1'
```

Retrieves the first 25 organizations. Data is retrieved in page mode to reduce network load and response times, and the retrieved organizations are sorted by their name in ascending order.

```
/Organizations?$skip=10&$top=20
```

Retrieves 20 organizations from the database where the first 10 organizations are omitted. That is, the retrieved organizations are those with sequence numbers 11 through 30.

```
/Organizations?$filter=organizationName/organizationName1 eq 'SAP SE'
```

Retrieves all organizations from the database where the first of the two name fields holds the string 'SAP SE'.

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Reason
200	Organization or Organizations retrieved successfully

## Payload

Format: *JSON*

Only media type JSON is supported. The JSON for this method has the following structure:

```
{ "basicData"      : {
  "businessPartnerID" : { "type": "string", "maxLength": 32},
  "tenant"            : { "type": "string", "maxLength": 36, "required":
true },
  "etag"              : { "type": "string" }
},
  "organizationName" : {
  "organizationName1" : { "type": "string", "maxLength": 255, "required":
true },
  "organizationName2" : { "type": "string", "maxLength": 255 }
},
  "communicationData" : {
  "emailAddress"      : { "type": "string", "maxLength": 255, "required":
true},
  "mobileDestinationLocation":{ "type": "string", "maxLength": 2 },
  "mobilePhoneNumber"   : { "type": "string", "maxLength": 127 },
  "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
  "landlinePhoneNumber" : { "type": "string", "maxLength": 127 },
  "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
  "streetName"          : { "type": "string", "maxLength": 255 },
  "houseNumber"         : { "type": "string", "maxLength": 10 },
  "cityName"            : { "type": "string", "maxLength": 255 },
  "postalCode"          : { "type": "string", "maxLength": 10 },
  "country"             : { "type": "string", "maxLength": 2 },
  "countryDescription"  : { "type": "string", "maxLength": 60, "readOnly" :
true },
  "region"              : { "type": "string", "maxLength": 3 },
  "regionDescription"   : { "type": "string", "maxLength": 60, "readOnly" :
true},
  "longitude"           : { "type": "number" },
  "latitude"            : { "type": "number" }
},
  "objectGroup"        : { "type": "string", "maxLength": 32, "required":
"true" },
  "externalIDs"         : {
    value: [{ "type": "string", "maxLength": 60,
"required": "false"}]
  }
}
```

## Related Information

[Organization \[page 28\]](#)

### 2.1.4 Read all Matching Organizations

Retrieves a list of `Organization` objects matching the filter criteria provided

With this method, you send a request to the server to retrieve a subset of all organizations, according to the filter criteria provided. This method offers an advanced filter functionality that goes beyond the standard filters used for collective requests of most of the other services available in the SAP IoT Application Services platform. Here, you can define a filter criterion that consists of a comma-separated list of individual organization IDs. The method then returns a collection of data sets for each `Organization` object specified in the request. This method is especially useful for administrative tasks affecting many users or things that are assigned to different organizations in the SAP IoT Application Services platform. For example, you might want to assign new expiration dates for the users assigned to all persons that belong to certain organizations.

## Request

**URI:** `/Organizations`

**HTTP Method:** [GET](#)

**Permissions:**

`<bp>.r`

`<bp>.tenant.r`

## Query String Parameters

Parameter	Required	Data Type	Description
<code>listFilterID</code>	Yes	string	comma-separated list of IDs referring to the <code>businessPartnerID</code> attribute of a <code>Organization</code>

## Request Example

```
/Organizations?
listFilterID=2c08c0b71187498498b113bf340186f8,2c08c0b71187498498b113bf340186f9,2c
08c0b71187498498b113bf340186fa
```

Retrieves three `Organization` objects with the specified IDs from the database.

## Response

### Response Status and Error Codes

Code	Description
200	List of organizations retrieved successfully
400	Bad request (for example, invalid filter)
403	Forbidden (user not authorized to access organizations)

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method is an array of multiple occurrences of the following structure (one for each retrieved organization):

```
{ "basicData"      : {
  "businessPartnerID" : { "type": "string", "maxLength": 32 },
  "tenant"            : { "type": "string", "maxLength": 36, "required":
true },
  "etag"              : { "type": "string" }
},
  "organizationName" : {
  "organizationName1" : { "type": "string", "maxLength": 255, "required":
true },
  "organizationName2" : { "type": "string", "maxLength": 255 }
},
  "communicationData" : {
  "emailAddress"      : { "type": "string", "maxLength": 255, "required":
true },
  "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
  "mobilePhoneNumber"  : { "type": "string", "maxLength": 127 },
  "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
  "landlinePhoneNumber" : { "type": "string", "maxLength": 127 },
  "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
  "streetName"         : { "type": "string", "maxLength": 255 },
  "houseNumber"        : { "type": "string", "maxLength": 10 },
  "cityName"           : { "type": "string", "maxLength": 255 },
  "postalCode"         : { "type": "string", "maxLength": 10 },
  "country"            : { "type": "string", "maxLength": 2 },
  "countryDescription"  : { "type": "string", "maxLength": 60, "readOnly" :
true },
  "region"             : { "type": "string", "maxLength": 3 },
  "regionDescription"   : { "type": "string", "maxLength": 60, "readOnly" :
true },
  "longitude"          : { "type": "number" },
  "latitude"           : { "type": "number" }
},
  "objectGroup"       : { "type": "string", "maxLength": 32, "required":
true },
  "externalIDs"        : {
    value: [{ "type": "string", "maxLength": 60,
"required": "false" }]
  }
}
```

## Related Information

[Organization \[page 28\]](#)

### 2.1.5 Update an Organization

With this method, you modify an organization with the specified ID

In the payload of the request, you assign values to those fields that you want to modify. For example, to add an organization name in addition to the mandatory `<organizationName1>`, you may pass a value for `<organizationName2>` in the payload of the service request. You could use this secondary name, for example, to differentiate between different departments within the same company.

#### Request

**URI:** `/Organizations(<businessPartnerID>)`

**HTTP Method:** *PUT*

**Permissions:**

`<bp>.r`

`<bp>.u`

`<bp>.tenant.r`

#### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross site request forgery token
If-Match	Yes	Check if cached version matches server version of the organization

#### Request Parameters

Name	Required	Type	Description	Send As
businessPartnerID	Yes	string	ID of the organization	Path parameter

#### Request Example

```
/Organizations('2c08c0b71187498498b113bf340186f8')
```

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method has the following structure:

```
{ "basicData"      : {
  "businessPartnerID" : { "type": "string", "maxLength": 32 },
  "tenant"            : { "type": "string", "maxLength": 36, "required":
true },
  "etag"              : { "type": "string" }
},
  "organizationName" : {
  "organizationName1" : { "type": "string", "maxLength": 255, "required":
true },
  "organizationName2" : { "type": "string", "maxLength": 255 }
},
  "communicationData" : {
  "emailAddress"      : { "type": "string", "maxLength": 255, "required":
true },
  "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
  "mobilePhoneNumber"   : { "type": "string", "maxLength": 127 },
  "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
  "landlinePhoneNumber" : { "type": "string", "maxLength": 127 },
  "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
  "streetName"          : { "type": "string", "maxLength": 255 },
  "houseNumber"         : { "type": "string", "maxLength": 10 },
  "cityName"            : { "type": "string", "maxLength": 255 },
  "postalCode"          : { "type": "string", "maxLength": 10 },
  "country"             : { "type": "string", "maxLength": 2 },
  "countryDescription"   : { "type": "string", "maxLength": 60, "readOnly" :
true },
  "region"              : { "type": "string", "maxLength": 3 },
  "regionDescription"    : { "type": "string", "maxLength": 60, "readOnly" :
true },
  "longitude"           : { "type": "number" },
  "latitude"            : { "type": "number" }
},
  "objectGroup"        : { "type": "string", "maxLength": 32, "required":
true },
  "externalIDs"        : {
    value: [{ "type": "string", "maxLength": 60,
"required": "false" }]
  }
}
```

## Response

Format: [JSON](#)

### Response Header Fields

Name	Type	Description
ETag	string	Identifier of the updated Organization instance version

### Response Status and Error Codes

Code	Reason
200	Organization updated successfully

## Related Information

[Organization \[page 28\]](#)

## 2.1.6 Delete an Organization

With this method, you delete an Organization with the specified ID.

With this method, you delete the organization with the specified ID from the system. If an organization is deleted, all assigned relationships to other business partners remain untouched, thus leaving the system in an inconsistent state. To prevent such inconsistencies, choose one of the following approaches:

- Before deleting an organization, check whether any relationships to other objects exist. Start with deleting these relationships, and then delete the organization.
- Use the onboarding services for organizations instead. These services are specially designed to handle not only the organization object, but also its relationships to other objects. For details, see [Delete Organization Assignment \[page 293\]](#).

### i Note

Deleting an organization is done in the form of a logical deletion only. That is, from a user's perspective, the deleted organization is no longer visible and cannot be accessed by the user. However, the physical organization object in the database remains untouched and can be retrieved by administrators, if necessary. This can be important, for example, for auditing purposes.

If the organization is successfully deleted, the server sends HTTP response code 204. Any other code indicates an error.

## Request

**URI:** `/Organizations('<businessPartnerID>')`

**HTTP Method:** `DELETE`

**Permissions:**

`<bp>.d`

`<bp>.r`

`<bp>.tenant.r`

**Request Header Fields**

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token
If-Match	Yes	Check if cached version matches server version of the organization

#### Request Parameters

Name	Required	Type	Description	Send As
businessPartnerID	Yes	string	technical ID of the Organization	Path parameter

#### Request Example

```
/Organizations('2c08c0b71187498498b113bf340186f8')
```

## Response

#### Response Status and Error Codes

Code	Reason
204	Organization deleted successfully

## Related Information

[Organization \[page 28\]](#)

[DeletedOrganization \[page 48\]](#)

## 2.1.7 Payload of CapabilityType "Organization"

Presents the payload structure of CapabilityType "Organization"

The payload of the CapabilityType service consists of two main parts:

- **vocabulary:** This part mirrors the payload elements of the Organization service. Any values that you enter here limit the scope of objects to those organizations that match the passed values.
- **actions:** This part defines which of the three possible types of action (read, write delete) shall be allowed for the Organization objects specified in the vocabulary.



## Payload

```
{
  "ID": "FFFFFFFFF00000000FFFFFFFFF0002",
  "name": "bp:organization",
  "vocabulary": {
    "attributes": [
      {
        "path": {
          "field": "cityName",
          "dimension": "root"
        },
        "valueType": {
          "type": "String",
          "length": 127
        }
      },
      {
        "path": {
          "field": "country",
          "dimension": "root"
        },
        "valueType": {
          "type": "String",
          "length": 127
        }
      },
      {
        "path": {
          "field": "postalCode",
          "dimension": "root"
        },
        "valueType": {
          "type": "String",
          "length": 127
        }
      },
      {
        "path": {
          "field": "organizationName1",
          "dimension": "root"
        },
        "valueType": {
          "type": "String",
          "length": 127
        }
      },
      {
        "path": {
          "field": "latitude",
          "dimension": "root"
        },
        "valueType": {
          "type": "NumericDecimal",
          "precision": 10,
          "scale": 3
        }
      },
      {
        "path": {
          "field": "mobileDestinationLocation",
          "dimension": "root"
        },
        "valueType": {
          "type": "String",
          "length": 127
        }
      }
    ]
  }
}
```

```

    },
    {
      "path": {
        "field": "emailAddress",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "landlineDestinationLocation",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "houseNumber",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "longitude",
        "dimension": "root"
      },
      "valueType": {
        "type": "NumericDecimal",
        "precision": 10,
        "scale": 3
      }
    },
    {
      "path": {
        "field": "district",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "landlinePhoneNumberExtension",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "objectGroup",
        "dimension": "root"
      },
    },

```

```

        "valueType": {
            "type": "HierarchyAssignment"
        }
    },
    {
        "path": {
            "field": "organizationName2",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "streetName",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "region",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "ID",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    }
]
},
"actions": {
    "actions": [
        {
            "name": "Read"
        },
        {
            "name": "Write"
        },
        {
            "name": "Delete"
        }
    ]
}
}

```

## Related Information

[CapabilityType \[page 221\]](#)

## 2.1.8 DeletedOrganization

A `DeletedOrganization` object is a physical backup for a logically deleted `Organization` object.

Whenever a user deletes an `Organization` object from an IoT scenario, this deletion is a logical rather than a physical deletion. That is, from a user's perspective, the deleted organization is no longer available in the system and cannot be accessed anymore. However, in the background, the system takes several steps to ensure that the deleted organization can still be restored, if need be. Here are some use cases where access to a previously deleted organization may be required:

- Restoring an organization that has accidentally been deleted
- Retrieving an organization for auditing purposes
- Retrieving an organization in the context of a lawsuit

Due to the special nature of the `DeletedOrganization` object, access to objects of this kind is limited to system administrators with special authorizations. Moreover, a `DeletedOrganization` object has a restricted set of methods that can be accessed via the Leonardo IoT API. For example, there is no `Create` method available, nor can it be modified. This is to make sure that a `DeletedOrganization` objects can only be created in the system according to certain predefined and hard-coded rules. These rules enforce that `DeletedOrganization` objects can only exist under well-defined circumstances laid down in legal frameworks that are relevant for operating an IoT scenario.

Write access to `DeletedOrganization` objects residing in a tenant other than the one to which a user is logged on is only allowed for users who are acting in the role of a cloud platform provider.

### Base URI:

- Formal description: `https://<server address>[:<port number>]/[<path>]/DeletedOrganizations`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/DeletedOrganizations`

## Methods

HTTP Method	Action	URI	Scopes
<i>GET</i>	<a href="#">Read a DeletedOrganization [page 50]</a>	<code>/DeletedOrganizations ('&lt;businessPartnerID&gt;')</code>	<code>&lt;bp&gt;.pd</code>
<i>GET</i>	<a href="#">Read all DeletedOrganizations [page 52]</a>	<code>/DeletedOrganizations</code>	<code>&lt;bp&gt;.pd</code>
<i>DELETE</i>	<a href="#">Delete a DeletedOrganization [page 53]</a>	<code>/DeletedOrganizations ('&lt;businessPartnerID&gt;')</code>	<code>&lt;bp&gt;.pd</code>

## i Note

As already mentioned above, the `DeletedOrganization` object has neither a POST nor a PUT method:

- There is no way of explicitly creating a `DeletedOrganization` object by a method call via the interface. Instead, a `DeletedOrganization` object is created automatically by the system whenever an `Organization` object is deleted.
- Once the system has created a `DeletedOrganization` object, there is no way of modifying its field values, neither explicitly nor implicitly. This is due to the strict legal requirements that are applicable in the use cases outlined above.

## Payload

Format: *JSON*

Only media type JSON is supported. The JSON for this method has the following structure:

```
{ "basicData"      : {
  "businessPartnerID" : { "type": "string", "maxLength": 32},
  "tenant"            : { "type": "string", "maxLength": 36, "required":
true },
  "etag"              : { "type": "string"
},
  "organizationName"  : {
  "organizationName1" : { "type": "string", "maxLength": 255, "required":
true },
  "organizationName2" : { "type": "string", "maxLength": 255 }
},
  "communicationData" : {
  "emailAddress"      : { "type": "string", "maxLength": 255, "required":
true},
  "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
  "mobilePhoneNumber"   : { "type": "string", "maxLength": 127 },
  "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
  "landlinePhoneNumber" : { "type": "string", "maxLength": 127 },
  "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
  "streetName"          : { "type": "string", "maxLength": 255 },
  "houseNumber"         : { "type": "string", "maxLength": 10 },
  "cityName"            : { "type": "string", "maxLength": 255 },
  "postalCode"          : { "type": "string", "maxLength": 10 },
  "country"             : { "type": "string", "maxLength": 2 },
  "countryDescription"  : { "type": "string", "maxLength": 60, "readOnly" :
true },
  "region"              : { "type": "string", "maxLength": 3 },
  "regionDescription"   : { "type": "string", "maxLength": 60, "readOnly" :
true},
  "longitude"           : { "type": "number" },
  "latitude"            : { "type": "number" }
},
  "objectGroup"        : { "type": "string", "maxLength": 32, "required":
"true" },
  "externalIDs"        : {
    value: [{ "type": "string", "maxLength": 60,
"required": "false"}]
  }
}
```

## Related Information

[Organization \[page 28\]](#)

[Business Partner \[page 25\]](#)

### 2.1.8.1 Read a DeletedOrganization

Retrieves the specified DeletedOrganization object from the database

With this method, you send a request to the server to retrieve a particular DeletedOrganization object specified by its ID.

If the DeletedOrganization object is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

## Request

**URI:** /DeletedOrganizations('businessPartnerID')

**HTTP Method:** *GET*

**Permissions:** <bp>.pd

## Request Header Fields

Name	Required	Description
If-Match	No	Check if cached version matches server version of the organization

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPartnerID	Yes	string	Technical ID of the DeletedOrganization object	Path

## Request Example

```
/DeletedOrganizations('84c8c41dcf8e4e139ce03d337ee14f8a')
```

## Response

### Format

**Format:** *JSON*

## Response Status and Error Codes

Code	Reason
200	DeletedOrganization retrieved successfully

## Payload

Format: *JSON*

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "basicData" : {
    "businessPartnerID": { "type": "string", "maxLength": 32 },
    "tenant"           : { "type": "string", "maxLength": 36 },
    "etag"             : { "type": "string" }
  },
  "personName": {
    "formOfAddress"      : { "type": "string", "maxLength": 10 },
    "formOfAddressDescription": { "type": "string", "maxLength": 30, "readOnly" :
true },
    "familyName"         : { "type": "string", "maxLength": 255, "required": true },
    "additionalFamilyName": { "type": "string", "maxLength": 255 },
    "familyNamePrefix": { "type": "string", "maxLength": 10 },
    "middleName": { "type": "string", "maxLength": 255 },
    "givenName": { "type": "string", "maxLength": 255, "required": true },
    "initials": { "type": "string", "maxLength": 10 },
    "academicTitle": { "type": "string", "maxLength": 4 },
    "academicTitleDescription": { "type": "string", "maxLength": 60, "readOnly" :
true },
    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,
"readOnly" : true }
  },
  "communicationData" : {
    "emailAddress": { "type": "string", "maxLength": 255, "required": true },
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
    "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
    "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
    "streetName": { "type": "string", "maxLength": 255 },
    "houseNumber": { "type": "string", "maxLength": 10 },
    "cityName": { "type": "string", "maxLength": 255 },
    "postalCode": { "type": "string", "maxLength": 10 },
    "country": { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60, "readOnly" : true },
    "region": { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
  },
  "objectGroup"      : { "type": "string", "maxLength": 32 }
}
```

## Related Information

[DeletedOrganization \[page 48\]](#)

## 2.1.8.2 Read all DeletedOrganizations

Retrieves a list of deleted organizations from the database

With this method, you send a request to the server to retrieve a subset of all deleted organizations, according to the filter criteria provided. For a collective request for a set of deleted organizations, you have these options:

- You can define that the set of matching deleted organizations shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

### Request

**URI:** /DeletedOrganizations

**HTTP Method:** *GET*

**Permissions:** <bp>.pd

### Query String Parameters

Name	Required	Data Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set
\$orderby	No	string	Field name to be used for sorting the result set in ascending order
\$filter	No	string	Filter condition to be applied; valid fields are basicData/tenant, organizationName/organizationName1, organizationName/organizationName2, objectGroup, and roleCategory
\$count	No	integer	Returns the number of entries in the database, with filter conditions taken into account (if applicable)

Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
basicData/tenant, organizationName/organizationName1, organizationName/organizationName2, objectGroup, roleCategory	eq, ne

### Request Example

```
/DeletedOrganizations
```

Retrieves all deleted organizations that are available in the database.

```
/DeletedOrganizations?$top=25&$orderby='organizationName.organizationName1'
```



Retrieves the first 25 deleted organizations. Data is retrieved in page mode to reduce network load and response times, and the retrieved deleted organizations are sorted by their name in ascending order.

```
/DeletedOrganizations?$skip=10&$top=20
```

Retrieves 20 deleted organizations from the database where the first 10 deleted organizations are omitted. That is, the retrieved deleted organizations are those with sequence numbers 11 through 30.

## Response

### Response Status and Error Codes

Code	Reason
200	Deleted organizations retrieved successfully

## Related Information

[DeletedOrganization](#) [page 48]

### 2.1.8.3 Delete a DeletedOrganization

With this method, you physically delete an organization with the specified ID.

Although "deleting a deleted organization" may sound strange at first glance, this is exactly the purpose of this method. The DeletedOrganization concept represents an organization that has played an active part in the system in the past, but has been logically deleted. As opposed to the logical deletion of an Organization object, deleting a DeletedOrganization object means:

- **explicit** physical deletion of the DeletedOrganization object, and
- **implicit** physical deletion of the Organization object that is represented by the corresponding DeletedOrganization object

In other words, deleting a DeletedOrganization object is the ultimate and irreversible way of removing all information about a particular organization from the system.

#### i Note

Be very careful before executing this method. Since it is the very purpose of the DeletedOrganization concept for you to be able to reconstruct the information stored for an organization even after that organization has been deleted from the system, we highly recommend that you make yourself aligned with all the relevant departments in your company before you delete a DeletedOrganization object.

## Request

**URI:** /DeletedOrganizations('<businessPartnerID>')

**HTTP Method:** *DELETE*

**Permissions:** <bp>.pd

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token
If-Match	Yes	Check if cached version matches server version of the DeletedOrganization object

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPartnerID	Yes	string	technical ID of the DeletedOrganization object	Path

### Request Example

```
/DeletedOrganizations('84c8c41dcf8e4e139ce03d337ee14f8a')
```

## Response

### Response Status and Error Codes

Code	Reason
204	DeletedOrganization deleted successfully

## Related Information

[DeletedOrganization \[page 48\]](#)

[Delete an Organization \[page 43\]](#)

## 2.2 Person

A Person represents a business partner of type person that is assigned to one tenant.

Persons are used to model representations of human beings who are involved in designing, operating, or monitoring an IoT services infrastructure. For persons, SAP provides the following predefined roles:

- Employee
- Contact person

It is obvious that these predefined roles are very abstract and generic. For productive use, it might be recommendable to define additional roles that describe the responsibilities of a particular person more precisely. Here is a list of some examples of potential roles for persons involved in an IoT scenario:

- System administrator
- Machine operator
- Quality manager
- Service technician

The tenant is an attribute of the person that must be provided upon creation of a person. That is, to create a person with a reference to a given tenant, the tenant name must be valid, and the tenant must already exist in the system.

To identify a person, the `familyName`, the `givenName`, and the `emailAddress` attributes are required. For the name fields, it is obvious that there can be no unique values required, since people with the same name are a normal everyday phenomenon. Given that, of the three required fields only the `emailAddress` attribute must be unique within the scope of the tenant to which the person belongs.

### i Note

There is one specialty with respect to the `emailAddress` field. As already mentioned, the e-mail address must be unique within the scope of a tenant. However, the same e-mail address can be assigned to more than one person as long as there is only one undeleted person in the system with that address. If an e-mail address has been used by one or more persons in the past but these persons have all been logically deleted, then the e-mail address is available again for assigning it to a different person. For example, this mechanism allows for assigning different employees to a permanent, non-individual, generic e-mail address like `support@company.com`.

The same generic e-mail address can also be assigned to the organization to which that person belongs. That is, the requirement for e-mail addresses to be unique within a tenant is a bit relaxed in that respect that it is allowed to use the same e-mail address for one person **and** one organization within the same tenant (resulting in two occurrences of the same e-mail address in one tenant).

If a person is deleted, the system automatically deletes all assigned relationships between that person and other objects.

### i Note

The Person services described here are focused on the isolated entity of a single person. This is fine for accessing or modifying certain fields and attributes of a person. However, for onboarding a person (that is, creating a new person and setting up relationships to associated objects), we recommend using the services described in the [Onboarding Persons \[page 295\]](#) section.

## Prerequisites

### Base URI:

- Formal description: `https://<server address>[:<port number>]/[path]/Persons`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/Persons`

## Methods

HTTP Method	Action	URI	Scopes
<i>POST</i>	<a href="#">Create a Person [page 59]</a>	<code>/Persons</code>	<code>&lt;bp&gt;.c</code> <code>&lt;bp&gt;.tenant.r</code>
<i>GET</i>	<a href="#">Read a Person [page 62]</a>	<code>/Persons ('&lt;ID&gt;')</code>	<code>&lt;bp&gt;.c</code> <code>&lt;bp&gt;.tenant.r</code>
<i>GET</i>	<a href="#">Read all Persons [page 64]</a>	<code>/Persons</code>	<code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.tenant.r</code>
<i>GET</i>	<a href="#">Read all Matching Persons [page 67]</a>	<code>/Persons</code>	<code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.tenant.r</code>
<i>PUT</i>	<a href="#">Update a Person [page 69]</a>	<code>/Persons ('&lt;ID&gt;')</code>	<code>&lt;bp&gt;.u</code> <code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.tenant.r</code>
<i>DELETE</i>	<a href="#">Delete a Person [page 72]</a>	<code>/Persons ('&lt;ID&gt;')</code>	<code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.d</code> <code>&lt;bp&gt;.tenant.r</code>

## Payload

Format: *JSON*

For PUT, POST, and GET requests, only the media type JSON is supported. The JSON for these methods has the following structure:

```
{"basicData" : {  
  "businessPartnerID": { "type": "string", "maxLength": 32 },
```

```

"tenant"          : { "type": "string", "maxLength": 36, "required": true },
"etag"            : { "type": "string" }
},
"personName": {
  "formOfAddress"      : { "type": "string", "maxLength": 10 },
  "formOfAddressDescription": { "type": "string", "maxLength": 30,
"readOnly" : true },
  "familyName"         : { "type": "string", "maxLength": 255, "required": true },
  "additionalFamilyName": { "type": "string", "maxLength": 255 },
  "familyNamePrefix": { "type": "string", "maxLength": 10 },
  "middleName": { "type": "string", "maxLength": 255 },
  "givenName": { "type": "string", "maxLength": 255, "required": true },
  "initials": { "type": "string", "maxLength": 10 },
  "academicTitle": { "type": "string", "maxLength": 4 },
  "academicTitleDescription": { "type": "string", "maxLength": 60,
"readOnly" : true },
  "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
  "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,
"readOnly" : true }
},
"communicationData" : {
  "emailAddress": { "type": "string", "maxLength": 255, "required": true },
  "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
  "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
  "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
  "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
  "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
  "streetName": { "type": "string", "maxLength": 255 },
  "houseNumber": { "type": "string", "maxLength": 10 },
  "cityName": { "type": "string", "maxLength": 255 },
  "postalCode": { "type": "string", "maxLength": 10 },
  "country": { "type": "string", "maxLength": 2 },
  "countryDescription": { "type": "string", "maxLength": 60, "readOnly" :
true },
  "region": { "type": "string", "maxLength": 3 },
  "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
},
"objectGroup": { "type": "string", "maxLength": 32, "required": "true" },
"externalIDs": {
  value: [{ "type": "string", "maxLength": 60, "required":
>false"}]
}
}

```

### i Note

Although the `givenName` as well as the `familyName` fields are defined with a field length of 255 characters, this length can only be used partially if the Person shall be associated with a User object (which applies in most cases). Due to limitations of the underlying user administration layer, the following restrictions apply:

- `givenName`: 31 characters
- `familyName`: 63 characters

### i Note

A special mechanism has been implemented for the fields dealing with phone numbers in the communication data block of the payload with respect to the preceding numeric country code. Unlike manual dialing with a phone, the country code is **not** given as the standardized number for a country (for


example, 1 for USA, 49 for Germany). Instead, the system expects the alphabetical ISO country code as a prefix (for example, "US" for USA, "DE" for Germany). The following fields are affected:

- `mobileDestinationLocation`
- `landlineDestinationLocation`

This mechanism is in effect for the service request types POST, GET, and PUT. You may look up the required ISO country codes in the [Country Codes \[page 1244\]](#) section of this documentation.

### i Note

For the `emailAddress` field, the following requirements apply:

- The content of the field must be a valid and well-formed e-mail address according to the rules set forth in [IETF RFC 2822](#) .
- The content of the field is validated internally by SAP Leonardo IoT with the help of the e-mail validator provided by the Hibernate framework.

As a consequence, the following applies: Although the technical field length is defined as 255 characters, neither the mailbox part (before the @ sign) nor the domain name (after the @ sign, but before the dot ".") may exceed a maximum length of 64 characters.

For more information, see the [Hibernate documentation](#) .

## External ID

In the payload, there is a special field that needs some additional explanation: `externalID`. This field is intended to hold an identifying string other than the purely technical `businessPartnerID` field, which the system needs for reasons of database consistency. In contrast to that, the `externalID` field can be used for providing an additional semantic identifier that users may be accustomed to from their daily business experience. For a person, this field could be used to hold the employee ID that a company has assigned to that person during the hiring process.

### i Note

The following limitations apply:

- Although the `externalIDs` field has been implemented as an array, the system currently supports the assignment of no more than one external ID (or none).
- On database level, the `externalIDs` field is managed as a common attribute of both persons and organizations. Therefore, the values entered for this field must be unique not only within a tenant, but also across persons and organizations. In other words, it is not allowed to use a value for the `externalIDs` field of a person that has already been used for an organization and vice versa.

## Access Authorization for Persons

Objects of type `Person` are subject to the instance-based authorization concept that is implemented in SAP Leonardo IoT. For more information, see [Object Instance-based Authorization \[page 205\]](#). To grant access to a Person instance, a capability has to be derived from the predefined `CapabilityType` object for persons. For more information on the `CapabilityType` payload needed for this purpose, see [Payload of CapabilityType "Person" \[page 73\]](#).

## Related Information

[Onboarding Persons \[page 295\]](#)

[Business Partner \[page 25\]](#)

### 2.2.1 Create a Person

With this method, you create a new business partner of type Person. The system automatically assigns a unique ID to the new person.

To access, create change and delete persons you need to have the respective authorization. This instance authorization is realized through hierarchical ObjectGroups and ObjectGroupCapabilities for the object type `com.sap.appcore:person`.

A user can only operate on persons that are assigned to the same tenant to which the user is logged on. Only a user acting as the cloud platform provider has the permission to access persons in other tenants through capabilities that are inherited from the root node.

If the person is successfully created, the server sends HTTP response code 201. Any other code indicates an error.

## Request

**URI:** `/Persons`

**HTTP Method:** `POST`

**Permissions:**

`<bp>.c`

`<bp>.tenant.r`

## Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token

## Request Fields

Parameter	Required	Data Type	Description	Send As
tenant	Yes	string	If omitted, tenant of currently logged on user is used.	JSON
familyName	Yes	string	Last name of the person to be created.	JSON

Parameter	Required	Data Type	Description	Send As
givenName	Yes	string	First name of the person to be created.	JSON
emailAddress	Yes	string	An email address used to get in contact with the person.	JSON

## Request Example

```
/Persons
```

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "basicData" : {
    "businessPartnerID": { "type": "string", "maxLength": 32 },
    "tenant" : { "type": "string", "maxLength": 36, "required": true },
    "etag" : { "type": "string" }
  },
  "personName": {
    "formOfAddress" : { "type": "string", "maxLength": 10 },
    "formOfAddressDescription": { "type": "string", "maxLength": 30,
    "readOnly" : true },
    "familyName" : { "type": "string", "maxLength": 255, "required": true },
    "additionalFamilyName": { "type": "string", "maxLength": 255 },
    "familyNamePrefix": { "type": "string", "maxLength": 10 },
    "middleName": { "type": "string", "maxLength": 255 },
    "givenName": { "type": "string", "maxLength": 255, "required": true },
    "initials": { "type": "string", "maxLength": 10 },
    "academicTitle": { "type": "string", "maxLength": 4 },
    "academicTitleDescription": { "type": "string", "maxLength": 60,
    "readOnly" : true },
    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,
    "readOnly" : true }
  },
  "communicationData" : {
    "emailAddress": { "type": "string", "maxLength": 255, "required": true },
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
    "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
    "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
    "streetName": { "type": "string", "maxLength": 255 },
    "houseNumber": { "type": "string", "maxLength": 10 },
    "cityName": { "type": "string", "maxLength": 255 },
    "postalCode": { "type": "string", "maxLength": 10 },
    "country": { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60, "readOnly" :
    true },
    "region": { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
  },
  "objectGroup": { "type": "string", "maxLength": 32, "required": "true" },
  "externalIDs": {
```



```

    value: [{ "type": "string", "maxLength": 60, "required":
"false"}]
    }
}

```


### i Note

Although the `givenName` as well as the `familyName` fields are defined with a field length of 255 characters, this length can only be used partially if the Person shall be associated with a User object (which applies in most cases). Due to limitations of the underlying user administration layer, the following restrictions apply:

- `givenName`: 31 characters
- `familyName`: 63 characters

### i Note

For the `emailAddress` field, the following requirements apply:

- The content of the field must be a valid and well-formed e-mail address according to the rules set forth in [IETF RFC 2822](#) .
- The content of the field is validated internally by SAP Leonardo IoT with the help of the e-mail validator provided by the Hibernate framework.

As a consequence, the following applies: Although the technical field length is defined as 255 characters, neither the mailbox part (before the @ sign) nor the domain name (after the @ sign, but before the dot ".") may exceed a maximum length of 64 characters.

For more information, see the [Hibernate documentation](#) .

## Response

### Response Header Fields

Name	Type	Description
Location	string	Path to the newly created Person
ETag	string	Entity tag of the newly created Person instance

### Response Status and Error Codes

Code	Reason
201	Person successfully created

## Related Information

[Person \[page 55\]](#)

## 2.2.2 Read a Person

Retrieves a specified person from the database

With this method, you send a request to the server to retrieve a particular Person object specified by its ID.

If the person is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

### Request

**URI:** `/Persons (<ID>')`

**HTTP Method:** *GET*

**Permissions:**

`<bp>.r`

`<bp>.tenant.r`

### Request Header Fields

Name	Required	Description
If-Match	Yes	Check if cached version matches server version of the person

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPa rtnerID	Yes	string	Technical ID of the Person object	Path

### Request Example

```
/Persons ('adbb2c0fdb0a4b9e99327a52c41fc720')
```

### Response

#### Format

**Format:** *JSON*

### Response Status and Error Codes

Code	Reason
200	Person retrieved successfully

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method has the following structure:

```
{ "basicData" : {
  "businessPartnerID": { "type": "string", "maxLength": 32 },
  "tenant"           : { "type": "string", "maxLength": 36, "required": true },
  "etag"             : { "type": "string" }
},
"personName": {
  "formOfAddress"      : { "type": "string", "maxLength": 10 },
  "formOfAddressDescription": { "type": "string", "maxLength": 30,
"readOnly" : true },
  "familyName"         : { "type": "string", "maxLength": 255, "required": true },
  "additionalFamilyName": { "type": "string", "maxLength": 255 },
  "familyNamePrefix"   : { "type": "string", "maxLength": 10 },
  "middleName"         : { "type": "string", "maxLength": 255 },
  "givenName"          : { "type": "string", "maxLength": 255, "required": true },
  "initials"           : { "type": "string", "maxLength": 10 },
  "academicTitle"      : { "type": "string", "maxLength": 4 },
  "academicTitleDescription": { "type": "string", "maxLength": 60,
"readOnly" : true },
  "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
  "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,
"readOnly" : true }
},
"communicationData" : {
  "emailAddress": { "type": "string", "maxLength": 255, "required": true },
  "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
  "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
  "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
  "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
  "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
  "streetName": { "type": "string", "maxLength": 255 },
  "houseNumber": { "type": "string", "maxLength": 10 },
  "cityName": { "type": "string", "maxLength": 255 },
  "postalCode": { "type": "string", "maxLength": 10 },
  "country": { "type": "string", "maxLength": 2 },
  "countryDescription": { "type": "string", "maxLength": 60, "readOnly" :
true },
  "region": { "type": "string", "maxLength": 3 },
  "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
},
"objectGroup": { "type": "string", "maxLength": 32, "required": "true" },
"externalIDs": {
  value: [{ "type": "string", "maxLength": 60, "required":
"false"}]
}
}
```

## Related Information

[Person \[page 55\]](#)

## 2.2.3 Read all Persons

Retrieves a list of persons from the database

With this method, you send a request to the server to retrieve a subset of all persons, according to the filter criteria provided. For a collective request for a set of persons, you have these options:

- You can restrict the set of persons to be retrieved by filter criteria based on the person name. In fact, defining filter criteria is mandatory. The system does not accept an unrestricted request for a list of all persons in the database.
- You can define that the set of matching persons shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

### i Note

Retrieving high numbers of objects from the database with only one collective GET request can lead to high system load and a significant increase of response times. Therefore, we highly recommend limiting the maximum number of returned objects to no more than 500 per method call. This can easily be accomplished with the help of the `$top` and `$skip` query parameters.

To avoid unwanted performance decline, collective GET requests are only supported up to a maximum of 500 objects that can be retrieved with one single call. Therefore, in use cases where it is crucial for you to ensure that all existing objects have been retrieved, you need to verify this by setting up a loop construct in your source code to repeat the request until the service method returns no further objects.

## Request

URI: `/Persons`

HTTP Method: `GET`

Permissions:

`<bp>.r`

`<bp>.tenant.r`

### Query String Parameters

Name	Required	Type	Description
<code>\$skip</code>	No	integer	Number of the first n records to be excluded from the result set
<code>\$top</code>	No	integer	Number of records to include in the result set

Name	Required	Type	Description
\$orderby	No	string	Field name to be used for sorting the result set in ascending or descending order; valid fields are: basicData.businessPartnerID, basicData.etag, personName.formOfAddressDescription, personName.familyName, personName.additionalFamilyName, personName.familyNamePrefix, personName.middleName, personName.givenName, personName.initials, personName.academicTitleDescription, personName.additionalAcademicTitleDescription, communicationData.emailAddress, communicationData.streetName, communicationData.houseNumber, communicationData.cityName, communicationData.postalCode, communicationData.country, communicationData.region
\$filter	No	string	Filter condition to be applied; valid fields are personName/familyName, personName/ givenName, communicationData/ emailAddress, basicData/ businessPartnerID, objectGroup, and roleCategory
\$count	No	integer	Returns the number of entries in the database, with filter conditions taken into account (if applicable)

Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
personName/familyName, personName/ givenName, communicationData/emailAddress, basicData/businessPartnerID, objectGroup, and roleCategory	eq, ne

## Request Example

```
/Persons
```

Retrieves all persons that are available in the database. Note that this formally unrestricted request is still implicitly restricted to Person objects in the current tenant for which the logged on user has an instance-based access authorization.

```
/Persons?$top=25&$orderby='personName.familyName' desc
```

Retrieves the first 25 persons. Data is retrieved in page mode to reduce network load and response times, and the retrieved persons are sorted by their family name in descending order.

```
/Persons?$skip=10&$top=20
```

Retrieves 20 persons from the database where the first 10 persons are omitted. That is, the retrieved persons are those with sequence numbers 11 through 30.

```
/Persons?$filter=personName/familyName eq 'Smith'
```

Retrieves all persons from the database whose family name is "Smith".

## Response

### Response Status and Error Codes

Code	Reason
200	Persons retrieved successfully

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "basicData" : {
    "businessPartnerID": { "type": "string", "maxLength": 32 },
    "tenant"           : { "type": "string", "maxLength": 36, "required": true },
    "etag"             : { "type": "string" }
  },
  "personName": {
    "formOfAddress"      : { "type": "string", "maxLength": 10 },
    "formOfAddressDescription": { "type": "string", "maxLength": 30,
    "readOnly" : true },
    "familyName"         : { "type": "string", "maxLength": 255, "required": true },
    "additionalFamilyName": { "type": "string", "maxLength": 255 },
    "familyNamePrefix"   : { "type": "string", "maxLength": 10 },
    "middleName"         : { "type": "string", "maxLength": 255 },
    "givenName"          : { "type": "string", "maxLength": 255, "required": true },
    "initials"           : { "type": "string", "maxLength": 10 },
    "academicTitle"      : { "type": "string", "maxLength": 4 },
    "academicTitleDescription": { "type": "string", "maxLength": 60,
    "readOnly" : true },
    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,
    "readOnly" : true }
  },
  "communicationData" : {
    "emailAddress": { "type": "string", "maxLength": 255, "required": true },
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
    "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
    "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
  }
}
```

```

    "streetName": { "type": "string", "maxLength": 255 },
    "houseNumber": { "type": "string", "maxLength": 10 },
    "cityName": { "type": "string", "maxLength": 255 },
    "postalCode": { "type": "string", "maxLength": 10 },
    "country": { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60, "readOnly" :
true },
    "region": { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
  },
  "objectGroup": { "type": "string", "maxLength": 32, "required": "true" },
  "externalIDs": {
    value: [{ "type": "string", "maxLength": 60, "required":
"false"}]
  }
}

```

## Related Information

[Person \[page 55\]](#)

## 2.2.4 Read all Matching Persons

Retrieves a list of `Person` objects matching the filter criteria provided

With this method, you send a request to the server to retrieve a subset of all persons, according to the filter criteria provided. This method offers an advanced filter functionality that goes beyond the standard filters used for collective requests of most of the other services available in the SAP Leonardo IoT platform. Here, you can define a filter criterion that consists of a comma-separated list of individual person IDs. The method then returns a collection of data sets for each `Person` object specified in the request. This method is especially useful to populate a data table with all the relevant data available for a given selection of persons. For example, you might want to use this method to provide person data for an administrative user interface application.

## Request

**URI:** `/Persons`

**HTTP Method:** `GET`

**Permissions:**

`<bp>.r`

`<bp>.tenant.r`

## Query String Parameters

Parameter	Required	Data Type	Description
listFilterID	Yes	string	comma-separated list of IDs referring to the businessPartnerID attribute of a Person

## Request Example

```
/Persons?  
listFilterID=adbb2c0fdb0a4b9e99327a52c41fc720,adbb2c0fdb0a4b9e99327a52c41fc721,ad  
bb2c0fdb0a4b9e99327a52c41fc722
```

Retrieves three `Person` objects with the specified IDs from the database.

## Response

### Response Status and Error Codes

Code	Description
200	List of persons retrieved successfully
400	Bad request (for example, invalid filter)
403	Forbidden (user not authorized to access persons)

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method is an array of multiple occurrences of the following structure (one for each retrieved person):

```
{ "basicData" : {  
  "businessPartnerID": { "type": "string", "maxLength": 32 },  
  "tenant"           : { "type": "string", "maxLength": 36, "required": true },  
  "etag"             : { "type": "string" }  
},  
  "personName": {  
    "formOfAddress"           : { "type": "string", "maxLength": 10 },  
    "formOfAddressDescription": { "type": "string", "maxLength": 30,  
    "readOnly" : true },  
    "familyName"              : { "type": "string", "maxLength": 255, "required": true },  
    "additionalFamilyName": { "type": "string", "maxLength": 255 },  
    "familyNamePrefix": { "type": "string", "maxLength": 10 },  
    "middleName": { "type": "string", "maxLength": 255 },  
    "givenName": { "type": "string", "maxLength": 255, "required": true },  
    "initials": { "type": "string", "maxLength": 10 },  
    "academicTitle": { "type": "string", "maxLength": 4 },  
    "academicTitleDescription": { "type": "string", "maxLength": 60,  
    "readOnly" : true },  
    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },  
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,  
    "readOnly" : true }  
  }
```



```

    },
    "communicationData" : {
      "emailAddress": { "type": "string", "maxLength": 255, "required": true },
      "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
      "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
      "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
      "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
      "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
      "streetName": { "type": "string", "maxLength": 255 },
      "houseNumber": { "type": "string", "maxLength": 10 },
      "cityName": { "type": "string", "maxLength": 255 },
      "postalCode": { "type": "string", "maxLength": 10 },
      "country": { "type": "string", "maxLength": 2 },
      "countryDescription": { "type": "string", "maxLength": 60, "readOnly" :
true },
      "region": { "type": "string", "maxLength": 3 },
      "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
    },
    "objectGroup": { "type": "string", "maxLength": 32, "required": "true" },
    "externalIDs": {
      value: [{ "type": "string", "maxLength": 60, "required":
"false"}]
    }
  }
}

```

## Related Information

[Person \[page 55\]](#)

## 2.2.5 Update a Person

With this method, you modify a person with the specified ID

In the payload of the request, you assign values to those fields that you want to modify. For example, to add the birth name of a married female person in addition to the mandatory `<familyName>`, you may pass a value for `<additionalFamilyName>` in the payload of the service request. With that, you could differentiate between different persons who, at first sight, have the same name.

## Request

**URI:** `/Persons('<businessPartnerID>')`

**HTTP Method:** *PUT*

**Permissions:**

`<bp>.u`

`<bp>.r`

`<bp>.tenant.r`

## Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token
If-Match	Yes	Check if cached version matches server version of the person

## Request Parameters

Name	Required	Type	Description	Send As
businessPartnerID	Yes	string	ID of the person	Path parameter
rtnerID				

## Request Example

```
/Persons ('adbb2c0fdb0a4b9e99327a52c41fc720')
```

## Payload

Format: [JSON](#)

For PUT, POST, and GET requests, only the media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "basicData" : {
    "businessPartnerID": { "type": "string", "maxLength": 32 },
    "tenant" : { "type": "string", "maxLength": 36, "required": true },
    "etag" : { "type": "string" }
  },
  "personName": {
    "formOfAddress" : { "type": "string", "maxLength": 10 },
    "formOfAddressDescription": { "type": "string", "maxLength": 30,
"readOnly" : true },
    "familyName" : { "type": "string", "maxLength": 255, "required": true },
    "additionalFamilyName": { "type": "string", "maxLength": 255 },
    "familyNamePrefix": { "type": "string", "maxLength": 10 },
    "middleName": { "type": "string", "maxLength": 255 },
    "givenName": { "type": "string", "maxLength": 255, "required": true },
    "initials": { "type": "string", "maxLength": 10 },
    "academicTitle": { "type": "string", "maxLength": 4 },
    "academicTitleDescription": { "type": "string", "maxLength": 60,
"readOnly" : true },
    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,
"readOnly" : true }
  },
  "communicationData" : {
    "emailAddress": { "type": "string", "maxLength": 255, "required": true },
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
    "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
    "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
    "streetName": { "type": "string", "maxLength": 255 },
    "houseNumber": { "type": "string", "maxLength": 10 },
    "cityName": { "type": "string", "maxLength": 255 },
    "postalCode": { "type": "string", "maxLength": 10 },

```

```

    "country": { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60, "readOnly" :
true },
    "region": { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
  },
  "objectGroup": { "type": "string", "maxLength": 32, "required": "true" },
  "externalIDs": {
    value: [{ "type": "string", "maxLength": 60, "required":
"false"}]
  }
}

```

### Note

Although the `givenName` as well as the `familyName` fields are defined with a field length of 255 characters, this length can only be used partially if the Person shall be associated with a User object (which applies in most cases). Due to limitations of the underlying user administration layer, the following restrictions apply:

- `givenName`: 31 characters
- `familyName`: 63 characters

## Response

### Format

Format: *JSON*

### Response Header Fields

Name	Type	Description
Etag	string	Identifies the updated Person instance

### Response Status and Error Codes

Code	Reason
200	Person updated successfully

## Related Information

[Person \[page 55\]](#)

## 2.2.6 Delete a Person

With this method, you delete a Person with the specified ID.

With this method, you delete the person with the specified ID from the system. If a person is deleted, all assigned relationships to other business partners remain untouched, thus leaving the system in an inconsistent state. To prevent such inconsistencies, choose one of the following approaches:

- Before deleting a person, check whether any relationships to other objects exist. Start with deleting these relationships, and then delete the person.
- Use the onboarding services for persons instead. These services are specially designed to handle not only the person object, but also its relationships to other objects.

### i Note

Deleting a person is done in the form of a logical deletion only. That is, from a user's perspective, the deleted person is no longer visible and cannot be accessed by the user. However, the physical Person object in the database remains untouched and can be retrieved by administrators, if necessary. This can be important, for example, for auditing purposes.

From a technical perspective, calling this method results in the following behavior: The system creates a new instance of a DeletedPerson object and copies all field values of the deleted Person object to the new DeletedPerson object. That is, the Person is cloned into a DeletedPerson. The DeletedPerson object can be retrieved by an administrator with special authorizations.

If the person is successfully deleted, the server sends HTTP response code 204. Any other code indicates an error.

## Request

**URI:** /Persons('<businessPartnerID>')

**HTTP Method:** *DELETE*

**Permissions:**

<bp>.r

<bp>.d

<bp>.tenant.r

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token
If-Match	Yes	Check if cached version matches server version of the person

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPa rtnerID	Yes	string	technical ID of the Person object	Path

## Request Example

```
/Persons ('adbb2c0fdb0a4b9e99327a52c41fc720')
```

## Response

### Response Status and Error Codes

Code	Reason
204	Person deleted successfully

## Related Information

[Person \[page 55\]](#)

[DeletedPerson \[page 77\]](#)

## 2.2.7 Payload of CapabilityType "Person"

Presents the payload structure of CapabilityType "Person"

The payload of the `CapabilityType` service consists of two main parts:

- `vocabulary`: This part mirrors the payload elements of the `Person` service. Any values that you enter here limit the scope of objects to those persons that match the passed values.
- `actions`: This part defines which of the three possible types of action (read, write delete) shall be allowed for the `Person` objects specified in the `vocabulary`.

## Payload

```
{
  "ID": "FFFFFFFFF00000000FFFFFFFFF0001",
  "name": "bp:person",
  "vocabulary": {
    "attributes": [
      {
```

```

        "path": {
            "field": "cityName",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "familyNamePrefix",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "country",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "additionalFamilyName",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "postalCode",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "latitude",
            "dimension": "root"
        },
        "valueType": {
            "type": "NumericDecimal",
            "precision": 10,
            "scale": 3
        }
    },
    {
        "path": {
            "field": "formOfAddress",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    }

```

```

    },
    {
      "path": {
        "field": "initials",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "mobileDestinationLocation",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "emailAddress",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "landlineDestinationLocation",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "familyName",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "houseNumber",
        "dimension": "root"
      },
      "valueType": {
        "type": "String",
        "length": 127
      }
    },
    {
      "path": {
        "field": "longitude",
        "dimension": "root"
      },
      "valueType": {

```

```

        "type": "NumericDecimal",
        "precision": 10,
        "scale": 3
    },
    {
        "path": {
            "field": "district",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "landlinePhoneNumberExtension",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "objectGroup",
            "dimension": "root"
        },
        "valueType": {
            "type": "HierarchyAssignment"
        }
    },
    {
        "path": {
            "field": "middleName",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "givenName",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "streetName",
            "dimension": "root"
        },
        "valueType": {
            "type": "String",
            "length": 127
        }
    },
    {
        "path": {
            "field": "region",
            "dimension": "root"
        }
    }

```



```

    },
    "valueType": {
      "type": "String",
      "length": 127
    }
  },
  {
    "path": {
      "field": "ID",
      "dimension": "root"
    },
    "valueType": {
      "type": "String",
      "length": 127
    }
  }
]
},
"actions": {
  "actions": [
    {
      "name": "Read"
    },
    {
      "name": "Write"
    },
    {
      "name": "Delete"
    }
  ]
}
}

```

## Related Information

[CapabilityType \[page 221\]](#)

### 2.2.8 DeletedPerson

A `DeletedPerson` object is a physical backup for a logically deleted `Person` object.

Whenever a user deletes a `Person` object from an IoT scenario, this deletion is a logical rather than a physical deletion. That is, from a user's perspective, the deleted person is no longer available in the system and cannot be accessed anymore. However, in the background, the system takes several steps to ensure that the deleted person can still be restored, if need be. Here are some use cases where access to a previously deleted person may be required:

- Restoring a person that has accidentally been deleted
- Retrieving a person for auditing purposes
- Retrieving a person in the context of a lawsuit

Due to the special nature of the `DeletedPerson` object, access to objects of this kind is limited to system administrators with special authorizations. Moreover, a `DeletedPerson` object has a restricted set of methods that can be accessed via the Leonardo IoT API. For example, there is no `Create` method available, nor can it be modified. This is to make sure that a `DeletedPerson` objects can only be created in the system according to

certain predefined and hard-coded rules. These rules enforce that `DeletedPerson` objects can only exist under well-defined circumstances laid down in legal frameworks that are relevant for operating an IoT scenario.

Write access to `DeletedPerson` objects residing in a tenant other than the one to which a user is logged on is only allowed for users who are acting in the role of a cloud platform provider.

**Base URI:** `/DeletedPersons ('<businessPartnerID>')`

**Permissions:** `<bp>.pd`

**Base URI:**

- Formal description: `https://<server address>[:<port number>]/<path>/DeletedPersons`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/DeletedPersons`

## Methods

HTTP Method	Action	URI	Scopes
<i>GET</i>	<a href="#">Read a DeletedPerson [page 79]</a>	<code>/DeletedPersons ('&lt;businessPartnerID&gt;')</code>	<code>&lt;bp&gt;.pd</code>
<i>GET</i>	<a href="#">Read all DeletedPersons [page 81]</a>	<code>/DeletedPersons</code>	<code>&lt;bp&gt;.pd</code>
<i>DELETE</i>	<a href="#">Delete a DeletedPerson [page 84]</a>	<code>/DeletedPersons ('&lt;businessPartnerID&gt;')</code>	<code>&lt;bp&gt;.pd</code>

### Note

As already mentioned above, the `DeletedPerson` object has neither a POST nor a PUT method:

- There is no way of explicitly creating a `DeletedPerson` object by a method call via the interface. Instead, a `DeletedPerson` object is created automatically by the system whenever a `Person` object is deleted.
- Once the system has created a `DeletedPerson` object, there is no way of modifying its field values, neither explicitly nor implicitly. This is due to the strict legal requirements that are applicable in the use cases outlined above.

## Payload

Format: *JSON*

Only media type JSON is supported. The JSON for these methods has the following structure:

```
{ "basicData" : {  
  "businessPartnerID": { "type": "string", "maxLength": 32 },
```

```

    "tenant"          : { "type": "string", "maxLength": 36 },
    "etag"            : { "type": "string" }
  },
  "personName": {
    "formOfAddress"      : { "type": "string", "maxLength": 10 },
    "formOfAddressDescription": { "type": "string", "maxLength": 30, "readOnly" :
true },
    "familyName"         : { "type": "string", "maxLength": 255, "required": true },
    "additionalFamilyName": { "type": "string", "maxLength": 255 },
    "familyNamePrefix": { "type": "string", "maxLength": 10 },
    "middleName": { "type": "string", "maxLength": 255 },
    "givenName": { "type": "string", "maxLength": 255, "required": true },
    "initials": { "type": "string", "maxLength": 10 },
    "academicTitle": { "type": "string", "maxLength": 4 },
    "academicTitleDescription": { "type": "string", "maxLength": 60, "readOnly" :
true },
    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,
"readOnly" : true }
  },
  "communicationData" : {
    "emailAddress": { "type": "string", "maxLength": 255, "required": true },
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
    "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
    "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
    "streetName": { "type": "string", "maxLength": 255 },
    "houseNumber": { "type": "string", "maxLength": 10 },
    "cityName": { "type": "string", "maxLength": 255 },
    "postalCode": { "type": "string", "maxLength": 10 },
    "country": { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60, "readOnly" : true },
    "region": { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
  },
  "objectGroup"      : { "type": "string", "maxLength": 32 }
}

```

## Related Information

[Person \[page 55\]](#)

[Business Partner \[page 25\]](#)

### 2.2.8.1 Read a DeletedPerson

Retrieves the specified DeletedPerson object from the database

With this method, you send a request to the server to retrieve a particular DeletedPerson object specified by its ID.

If the DeletedPerson object is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

## Request

URI: /DeletedPersons ('<businessPartnerID>')

HTTP Method: [GET](#)

Permissions: <bp>.pd

### Request Header Fields

Name	Required	Description
If-Match	No	Check if cached version matches server version of the person

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPartnerID	Yes	string	Technical ID of the DeletedPerson object	Path

### Request Example

```
/DeletedPersons ('84c8c41dcf8e4e139ce03d337ee14f8a')
```

## Response

### Format

Format: [JSON](#)

### Response Status and Error Codes

Code	Reason
200	DeletedPerson retrieved successfully

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method has the following structure:

```
{ "basicData" : {
  "businessPartnerID": { "type": "string", "maxLength": 32 },
  "tenant"           : { "type": "string", "maxLength": 36 },
  "etag"             : { "type": "string" }
},
"personName": {
  "formOfAddress"      : { "type": "string", "maxLength": 10 },
```

```

    "formOfAddressDescription": { "type": "string", "maxLength": 30, "readOnly" :
true },
    "familyName" : { "type": "string", "maxLength": 255, "required": true },
    "additionalFamilyName": { "type": "string", "maxLength": 255 },
    "familyNamePrefix": { "type": "string", "maxLength": 10 },
    "middleName": { "type": "string", "maxLength": 255 },
    "givenName": { "type": "string", "maxLength": 255, "required": true },
    "initials": { "type": "string", "maxLength": 10 },
    "academicTitle": { "type": "string", "maxLength": 4 },
    "academicTitleDescription": { "type": "string", "maxLength": 60, "readOnly" :
true },
    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,
"readOnly" : true }
},
"communicationData" : {
    "emailAddress": { "type": "string", "maxLength": 255, "required": true },
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
    "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
    "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
    "streetName": { "type": "string", "maxLength": 255 },
    "houseNumber": { "type": "string", "maxLength": 10 },
    "cityName": { "type": "string", "maxLength": 255 },
    "postalCode": { "type": "string", "maxLength": 10 },
    "country": { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60, "readOnly" : true },
    "region": { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
},
"objectGroup" : { "type": "string", "maxLength": 32 }
}

```

## Related Information

[DeletedPerson \[page 77\]](#)

### 2.2.8.2 Read all DeletedPersons

Retrieves a list of deleted persons from the database

With this method, you send a request to the server to retrieve a subset of all deleted persons, according to the filter criteria provided. For a collective request for a set of deleted persons, you have these options:

- You can define that the set of matching deleted persons shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

## Request

**URI:** /DeletedPersons

**HTTP Method:** *GET*

Permissions: <bp>.pd

## Query String Parameters

Name	Required	Data Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set
\$orderby	No	string	Field name to be used for sorting the result set in ascending order
\$filter	No	string	Filter condition to be applied; valid fields are personName/familyName, personName/givenName, communicationData/emailAddress, basicData/businessPartnerID, objectGroup, and roleCategory
\$count	No	integer	Returns the number of entries in the database, with filter conditions taken into account (if applicable)

Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
personName/familyName, personName/givenName, communicationData/emailAddress, basicData/businessPartnerID, objectGroup, roleCategory	eq, ne

## Request Example

```
/DeletedPersons
```

Retrieves all deleted persons that are available in the database.

```
/DeletedPersons?$top=25&$orderby='familyName'
```

Retrieves the first 25 deleted persons. Data is retrieved in page mode to reduce network load and response times, and the retrieved deleted persons are sorted by their family name in ascending order.

```
/DeletedPersons?$skip=10&$top=20
```

Retrieves 20 deleted persons from the database where the first 10 deleted persons are omitted. That is, the retrieved deleted persons are those with sequence numbers 11 through 30.

## Response

### Response Status and Error Codes

Code	Reason
200	Deleted persons retrieved successfully

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "basicData" : {
    "businessPartnerID": { "type": "string", "maxLength": 32 },
    "tenant"           : { "type": "string", "maxLength": 36 },
    "etag"             : { "type": "string" }
  },
  "personName": {
    "formOfAddress"      : { "type": "string", "maxLength": 10 },
    "formOfAddressDescription": { "type": "string", "maxLength": 30, "readOnly" : true },
    "familyName"         : { "type": "string", "maxLength": 255, "required": true },
    "additionalFamilyName": { "type": "string", "maxLength": 255 },
    "familyNamePrefix": { "type": "string", "maxLength": 10 },
    "middleName": { "type": "string", "maxLength": 255 },
    "givenName": { "type": "string", "maxLength": 255, "required": true },
    "initials": { "type": "string", "maxLength": 10 },
    "academicTitle": { "type": "string", "maxLength": 4 },
    "academicTitleDescription": { "type": "string", "maxLength": 60, "readOnly" : true },
    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60, "readOnly" : true }
  },
  "communicationData" : {
    "emailAddress": { "type": "string", "maxLength": 255, "required": true },
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
    "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
    "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
    "streetName": { "type": "string", "maxLength": 255 },
    "houseNumber": { "type": "string", "maxLength": 10 },
    "cityName": { "type": "string", "maxLength": 255 },
    "postalCode": { "type": "string", "maxLength": 10 },
    "country": { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60, "readOnly" : true },
    "region": { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
  },
  "objectGroup"      : { "type": "string", "maxLength": 32 }
}
```

## Related Information

[DeletedPerson \[page 77\]](#)

### 2.2.8.3 Delete a DeletedPerson

With this method, you physically delete a Person with the specified ID.

Although "deleting a deleted person" may sound strange at first glance, this is exactly the purpose of this method. The DeletedPerson concept represents a person who has played an active part in the system in the past, but has been logically deleted. As opposed to the logical deletion of a Person object, deleting a DeletedPerson object means:

- **explicit** physical deletion of the DeletedPerson object, and
- **implicit** physical deletion of the Person object that is represented by the corresponding DeletedPerson object

In other words, deleting a DeletedPerson object is the ultimate and irreversible way of removing all information about a particular person from the system.

#### Note

Be very careful before executing this method. Since it is the very purpose of the DeletedPerson concept for you to be able to reconstruct the information stored for a person even after that person has been deleted from the system, we highly recommend that you make yourself aligned with all the relevant departments in your company before you delete a DeletedPerson object.

## Request

**URI:** /DeletedPersons ('<businessPartnerID>')

**HTTP Method:** *DELETE*

**Permissions:** <bp>.pd

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token
If-Match	Yes	Check if cached version matches server version of the DeletedPerson object



## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPartnerID	Yes	string	technical ID of the DeletedPerson object	Path

## Request Example

```
/DeletedPersons('84c8c41dcf8e4e139ce03d337ee14f8a')
```

## Response

### Response Status and Error Codes

Code	Reason
204	DeletedPerson deleted successfully

## Related Information

[DeletedPerson \[page 77\]](#)

[Delete a Person \[page 72\]](#)

## 2.3 Retrieving Enriched Person Data

Describes how to retrieve person and user data simultaneously

Whenever new tenants are to be onboarded to the Leonardo IoT platform, the responsible system administrator must usually handle a high number of data related to persons as well as system users for these persons. For example, when it comes to set up an administrative user interface for managing persons and users, the desire for a simultaneous access to both types of data arises. The standard "Read" methods of the basic entities in the data model cannot be used for this. Therefore, an additional type of "Read" methods has been implemented for certain services offering advanced filters, which, in combination with other services, help you fulfill the demand described above.

The following service methods are available for this use case:

- [Read all Matching Persons \[page 67\]](#)
- [Read all Matching Organizations \[page 39\]](#)
- [Read all Matching BPUserRelationships \[page 123\]](#)
- [Read all Matching Locations \[page 140\]](#)

## Usage Suggestions

Here is a collection of high-level examples of how you can proceed to implement a solution for the use case described above with the help of the enhanced read methods:

### Retrieve enriched person data using an organization ID as a starting point

1. Get all PersonIDs for the Organization (`/BPRelationships?$filter=businessPartner1ID eq <OrgID> or businessPartner2ID eq <OrgID> and relationshipType eq com.sap.appcore:BUR010`)
2. Create an ID list from the result retrieved in (1)
3. Get all Person Details for IDs from the list created in (2) (`Persons?listFilterID=<ID1>,<ID2>,<ID3>...`)
4. Get all users for IDs from the list created in (2) (`BPUserRelationships?listFilterID=<ID1>,<ID2>,<ID3>...`)

#### Note

For performance reasons, we recommend having the system process the previous two calls in parallel.

5. Get all user groups assigned to the users retrieved in (4). To accomplish this, you need to send a call to the identity provider. Currently, a GET request for a single usergroup is available (`service/um/<USERNAME>`).

### Retrieve enriched person data using a tenant as a starting point

1. Get all Person details for persons in tenant (`Persons?$filter=basicData/tenant eq <tenantName>`)
2. Create an ID list from the result retrieved in (1)
3. Get all users for the list created in (2) (`BPUserRelationships?listFilterID=<ID1>,<ID2>,<ID3>...`)
4. Get all user groups assigned to the users retrieved in (3). To accomplish this, you need to send a call to the identity provider. Currently, a GET request for a single usergroup is available (`service/um/<USERNAME>`).

### Retrieve enriched person data using user group as a starting point

1. Get all users for a user group (POST service on endpoint `service/um` with body `{ "group_name": "UserGroupName" }`)
2. Create a user name list from the result retrieved in (1)
3. Get all IDs for the list created in (2) (`BPUserRelationships?listFilterUserName=<Name1>,<Name2>,<Name3> ...`)
4. Create an ID list from the result retrieved in (3)
5. Get all person details for IDs from the list created in (4) (`Persons?listFilterID=<ID1>,<ID2>,<ID3>...`)
6. Only if you need additional user groups: Get all user groups assigned to the users retrieved in (2). For this, the identity provider has to be called. Currently, a GET request for a single usergroup is available (`service/um/<USERNAME>`).

## Related Information

[Tenant \[page 282\]](#)

[Organization \[page 28\]](#)

[Person \[page 55\]](#)

[BPUserRelationship \[page 116\]](#)

[Location \[page 134\]](#)

## 2.4 BPRelationship

Describes the relationship between two business partners

A BPRelationship object describes the relationship of one business partner (organization or person) to another business partner. Here are some examples of relationships between business partners:

- A is employee of B
- A is customer of B
- A is supplier of B

To fully describe the relationship between two business partners, the following data needs to be submitted:

- both business partners
- type of the relationship
- flag indicating whether a relationship is the default relationship

It is possible to set up more than one relationship between two business partners. For example, it can easily happen that two companies are connected to each other in a way where both companies act in the role as a supplier **and** a customer at the same time, but with regard to different products or services. In that case, it may be relevant for certain use cases to decide which of these relationships is most relevant. To make this clear, use the `isDefault` flag.

### Authorization

Write access to BPRelationship objects implies also changes to the business partner objects that are connected to each other via the relationship. These implicit changes are carried out automatically by the system. Therefore, for accessing a BPRelationship, a user needs the respective capabilities for the involved Person or Organization objects.

### Base URI:

- Formal description: `https://<server address>[:<port number>][<path>]/BPRelationships`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/BPRelationships`

## Payload

For GET and POST requests, only media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "relationshipID"      : { "type": "string", "maxLength": 32 },
  "tenant"              : { "type": "string", "maxLength": 36 },
  "businessPartner1ID"  : { "type": "string", "minLength": 32, "maxLength":
32, "required": true },
  "businessPartner2ID"  : { "type": "string", "minLength": 32, "maxLength":
32, "required": true },
  "relationshipType"     : { "type": "string", "maxLength": 50, "required":
true },
  "oneToTwoDescription" : { "type": "string", "maxLength": 60 },
  "twoToOneDescription" : { "type": "string", "maxLength": 60 },
  "isDefault"           : { "type": "integer", "maxLength": 1 }
}
```

### Field Explanation

Name	Description
relationshipID	Unique identifier of the relationship (automatically generated by the system)
tenant	Tenant to which all involved objects belong
businessPartner1ID, businessPartner2ID	IDs of the two related business partners
relationshipType	Type of the relationship. The following relationship types are available: <ul style="list-style-type: none"><li>com.sap.appcore: BUR010: Is Employee</li><li>com.sap.appcore: IOT001: Is Onboarding Organization</li><li>com.sap.appcore: BUR001: Is Contact Person</li></ul>
oneToTwoDescription, twoToOneDescription	Semantic description of the relationship (one for each business partner's perspective/read only)
isDefault	Indicates whether the relationship is the default relationship (in case additional relationships exist for the two business partners involved)

## Methods

HTTP Method	Action	URI	Scopes
POST	<a href="#">Create a BPRelationship [page 89]</a>	/BPRelationships	<bp>.c  <bp>.tenant.r

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read a BPRelationship [page 91]</a>	/BPRelationships('<relationshipID>')	<bp>.r <bp>.tenant.r
GET	<a href="#">Read all BPRelationships [page 92]</a>	/BPRelationships	<bp>.r <bp>.tenant.r
DELETE	<a href="#">Delete a BPRelationship [page 95]</a>	/BPRelationships('<relationshipID>')	<bp>.d <bp>.tenant.r

### Note

There is no method available for modifying the values of an already existing relationship. If changes are necessary, you can accomplish this by deleting the current relationship and create it anew with the changed field values.

## Related Information

[Business Partner \[page 25\]](#)

[Organization \[page 28\]](#)

[Person \[page 55\]](#)

## 2.4.1 Create a BPRelationship

With this method, you create a new relationship between two business partners

To create a new relationship, you need to provide the IDs of the two business partners, the relationship type and an indication whether the new relationship is the default relationship for the two business partners. All of these data must be provided via the respective payload fields of this method.

## Request

**URI:** /BPRelationships

**HTTP Method:** *POST*

**Permissions:**

<bp>.c

<bp>.tenant.r

## Request Header Fields

Name	Required	Values
X-CSRF-Token	Yes	Cross site request forgery token

## Request Parameters

None.

## Request Example

```
/BPRelationships
```

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "relationshipID"      : { "type": "string", "maxLength": 32 },
  "tenant"              : { "type": "string", "maxLength": 36 },
  "businessPartner1ID"  : { "type": "string", "minLength": 32, "maxLength":
32, "required": true },
  "businessPartner2ID"  : { "type": "string", "minLength": 32, "maxLength":
32, "required": true },
  "relationshipType"    : { "type": "string", "maxLength": 50, "required":
true },
  "oneToTwoDescription" : { "type": "string", "maxLength": 60 },
  "twoToOneDescription" : { "type": "string", "maxLength": 60 },
  "isDefault"           : { "type": "integer", "maxLength": 1 }
}
```

Field Explanation

Name	Description
relationshipID	Unique identifier of the relationship
tenant	Tenant to which all involved objects belong
businessPartner1ID, businessPartner2ID	IDs of the two related business partners
relationshipType	Type of the relationship. The following relationship types are available: <ul style="list-style-type: none"><li>com.sap.appcore: BUR010: Is Employee</li><li>com.sap.appcore: IOT001: Is Onboarding Organization</li><li>com.sap.appcore: BUR001: Is Contact Person</li></ul>
oneToTwoDescription, twoToOneDescription	Semantic description of the relationship (one for each business partner's perspective)

Name	Description
<code>isDefault</code>	Indicates whether the relationship is the default relationship (in case additional relationships exist for the business partners involved)

## Response

### Response Header Fields

Name	Description
<code>Location</code>	Path to the the newly created relationship

### Response Status and Error Codes

Code	Description
201	Relationship created successfully

## Related Information

[BPRelationship \[page 87\]](#)

## 2.4.2 Read a BPRelationship

Retrieves a specified relationship from the database

With this method, you send a request to the server to retrieve a particular BPRelationship object specified by its ID.

## Request

**URI:** `/BPRelationships('<relationshipID>')`

**HTTP Method:** [GET](#)

**Permissions:**

`<bp>.r`

`<bp>.tenant.r`

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
relationships ipID	Yes	string	Technical identifier of the relationship	Path

## Request Example

```
/BPRelationships('95bf365108dc49a78e9ce4c173e3acd6')
```

## Response

### Response Status and Error Codes

Code	Description
200	Relationship retrieved successfully

## Related Information

[BPRelationship \[page 87\]](#)

## 2.4.3 Read all BPRelationships

Retrieves a list of relationships from the database

With this method, you send a request to the server to retrieve a subset of all BPRelationships, according to the filter criteria provided. For a collective request for a set of relationships, you have these options:

- You can restrict the set of relationships to be retrieved by filter criteria based on the business partners involved.
- You can define that the set of matching relationships shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

### i Note

For this service method, the `$filter` parameter supports additional features compared to the same parameter available for most of the other services. You can combine several filter conditions with the help of the logical operators `AND` and `OR`. With that, you can search for a given business partner without having to know on which side of a relationship that business partner has been assigned. Also, it is possible to combine the search for a particular business partner with a restriction to a certain type of relationship (for example, "is employee of").



## i Note

Retrieving high numbers of objects from the database with only one collective GET request can lead to high system load and a significant increase of response times. Therefore, we highly recommend limiting the maximum number of returned objects to no more than 500 per method call. This can easily be accomplished with the help of the `$top` and `$skip` query parameters.

To avoid unwanted performance decline, collective GET requests are only supported up to a maximum of 500 objects that can be retrieved with one single call. Therefore, in use cases where it is crucial for you to ensure that all existing objects have been retrieved, you need to verify this by setting up a loop construct in your source code to repeat the request until the service method returns no further objects.

## Request

**URI:** `/BPRelationships`

**HTTP Method:** `GET`

**Permissions:**

`<bp>.r`

`<bp>.tenant.r`

## Query String Parameters

Name	Required	Data Type	Description
<code>\$skip</code>	No	integer	Number of the first n records to be excluded from the result set
<code>\$top</code>	No	integer	Number of records to include in the result set
<code>\$filter</code>	No	string	Filter condition to be applied; valid fields are <code>businessPartner1ID</code> , <code>businessPartner2ID</code> , and <code>relationshipType</code>
<code>\$count</code>	No	integer	Returns the number of entries in the database, with filter conditions taken into account (if applicable)

## i Note

Although from a technical perspective all the query parameters listed above are **not** mandatory, we still recommend restricting the result set of elements.

## i Note

If you want to use the restriction to a certain `relationshipType` in the filter condition, make sure that a restriction based on that field must be passed to the service either at the beginning or at the end of the entire filter condition. Any other position of this condition part will cause a parser error.

#### Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
businessPartner1ID, businessPartner2ID, relationshipType	eq, ne

### Request Example

```
/BPRelationships
```

Retrieves all relationships that are available in the database.

```
/BPRelationships?$top=25
```

Retrieves the first 25 relationships. Data is retrieved in page mode to reduce network load and response times.

```
/BPRelationships?$filter=businessPartner1ID eq '95bf365108dc49a78e9ce4c173e3acd6'
```

Retrieves all relationships where the business partner with the ID 95bf365108dc49a78e9ce4c173e3acd6 is mentioned as the first of the two involved business partners.

```
/BPRelationships?$filter=businessPartner1ID eq '95bf365108dc49a78e9ce4c173e3acd6'
or
'95bf365108dc49a78e9ce4c173e3acd6' and businessPartner2ID eq
relationshipType eq 'com.sap.appcore:BUR010'
```

Retrieves all relationships where the business partner with the ID 95bf365108dc49a78e9ce4c173e3acd6 is assigned to either side of a relationship, as long as the relationship type is "is employee".

## Response

### Response Status and Error Codes

Code	Description
200	List of relationships retrieved successfully
400	Bad request (for example, invalid filter)
403	Forbidden (user not authorized to access relationships)

## Related Information

[BPRelationship \[page 87\]](#)

## 2.4.4 Delete a BPRelationship

With this method, you delete a BPRelationship with the specified ID.

Deleting a relationship is always necessary when you plan to delete one of the two business partners that are connected via that relationship. Otherwise, the remaining business partner would continue to carry the information about being related to the now deleted business partner, thus bringing the system into an inconsistent state. There is no automatic cascading of delete information to the affected objects.

### Request

**URI:** /BPRelationships('<relationshipID>')

**HTTP Method:** *DELETE*

**Permissions:**

<bp>.d

<bp>.tenant.r

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross site request forgery token

### Request Parameters

None.

### Request Example

```
/BPRelationships('95bf365108dc49a78e9ce4c173e3acd6')
```

### Response

#### Response Status and Error Codes

Code	Description
204	Relationship deleted successfully

### Related Information

[BPRelationship \[page 87\]](#)

## 2.4.5 DeletedBPRelationship

A DeletedBPRelationship object is a physical backup for a logically deleted BPRelationship object.

Whenever a user deletes a relationship between two business partners from an IoT scenario, this deletion is a logical rather than a physical deletion. That is, from a user's perspective, the deleted relationship is no longer available in the system and cannot be accessed anymore. However, in the background, the system takes several steps to ensure that the deleted relationship can still be restored, if need be. Here are some use cases where access to a previously deleted relationship may be required:

- Restoring a relationship that has accidentally been deleted
- Retrieving a relationship for auditing purposes
- Retrieving a relationship in the context of a lawsuit

Due to the special nature of the DeletedBPRelationship object, access to objects of this kind is limited to system administrators with special authorizations. Moreover, a DeletedBPRelationship object has a restricted set of methods that can be accessed via the Leonardo IoT API. For example, there is no Create method available, nor can it be modified. This is to make sure that a DeletedBPRelationship objects can only be created in the system according to certain predefined and hard-coded rules. These rules enforce that DeletedBPRelationship objects can only exist under well-defined circumstances laid down in legal frameworks that are relevant for operating an IoT scenario.

Write access to DeletedBPRelationship objects residing in a tenant other than the one to which a user is logged on is only allowed for users who are acting in the role of a cloud platform provider.

**Base URI:** /DeletedBPRelationship('<relationshipID>')

**Permissions:** <bp>.pd

**Base URI:**

- Formal description: `https://<server address>[:<port number>]/[<path>]/DeletedBPRelationships`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/DeletedBPRelationships`

## Methods

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read a DeletedBPRelationship [page 98]</a>	/DeletedBPRelationships('<relationshipID>')	<bp>.pd
GET	<a href="#">Read all DeletedBPRelationships [page 99]</a>	/DeletedBPRelationships	<bp>.pd

HTTP Method	Action	URI	Scopes
<b>DELETE</b>	<a href="#">Delete a DeletedBPRelationship</a> <a href="#">[page 101]</a>	/ DeletedBPRelationships ('<relationshipID>')	<bp>.pd

## i Note

As already mentioned above, the DeletedBPRelationship object has neither a POST nor a PUT method:

- There is no way of explicitly creating a DeletedBPRelationship object by a method call via the interface. Instead, a DeletedBPRelationship object is created automatically by the system whenever a BPRelationship object is deleted.
- Once the system has created a DeletedBPRelationship object, there is no way of modifying its field values, neither explicitly nor implicitly. This is due to the strict legal requirements that are applicable in the use cases outlined above.

## Payload

Only media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "relationshipID"      : { "type": "string", "maxLength": 32 },
  "tenant"              : { "type": "string", "maxLength": 36 },
  "businessPartner1ID"  : { "type": "string", "minLength": 32, "maxLength": 32, "required": true },
  "businessPartner2ID"  : { "type": "string", "minLength": 32, "maxLength": 32, "required": true },
  "relationshipType"     : { "type": "string", "maxLength": 50, "required": true },
  "oneToTwoDescription" : { "type": "string", "maxLength": 60 },
  "twoToOneDescription" : { "type": "string", "maxLength": 60 },
  "isDefault"           : { "type": "integer", "maxLength": 1 }
}
```

Field Explanation

Name	Description
relationshipID	Unique identifier of the relationship (automatically generated by the system)
tenant	Tenant to which all involved objects belong
businessPartner1ID, businessPartner2ID	IDs of the two related business partners
relationshipType	Type of the relationship. The following relationship types are available: <ul style="list-style-type: none"> <li>• com.sap.appcore: BUR010: Is Employee</li> <li>• com.sap.appcore: IOT001: Is Onboarding Organization</li> <li>• com.sap.appcore: BUR001: Is Contact Person</li> </ul>

Name	Description
oneToTwoDescription, twoToOneDescription	Semantic description of the relationship (one for each business partner's perspective/read only)
isDefault	Indicates whether the relationship is the default relationship (in case additional relationships exist for the two business partners involved)

## Related Information

[Read a DeletedBPRelationship \[page 98\]](#)

[Read all DeletedBPRelationships \[page 99\]](#)

[Delete a DeletedBPRelationship \[page 101\]](#)

### 2.4.5.1 Read a DeletedBPRelationship

Retrieves the specified DeletedBPRelationship object from the database

With this method, you send a request to the server to retrieve a particular DeletedBPRelationship object specified by its ID.

If the DeletedBPRelationship object is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

## Request

**URI:** /DeletedBPRelationships('<relationshipID>')

**HTTP Method:** *GET*

**Permissions:** <bp>.pd

## Request Header Fields

Name	Required	Description
If-Match	No	Check if cached version matches server version of the relationship

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
relationshipID	Yes	string	Technical ID of the DeletedBPRelationship object	Path

## Request Example

```
/DeletedBPRelationships('9828317b1aee41e391aeb5cdd3b4fb6d')
```

## Response

### Format

Format: *JSON*

### Response Status and Error Codes

Code	Reason
200	DeletedBPRelationship retrieved successfully

## Related Information

[DeletedBPRelationship \[page 96\]](#)

## 2.4.5.2 Read all DeletedBPRelationships

Retrieves a list of deleted relationships between business partners from the database

With this method, you send a request to the server to retrieve a subset of all deleted relationships between business partners (persons or organizations), according to the filter criteria provided. For a collective request for a set of deleted relationships, you have these options:

- You can define that the set of matching deleted relationships shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

## Request

URI: `/DeletedBPRelationships`

HTTP Method: *GET*

Permissions: <bp>.pd

## Query String Parameters

Name	Required	Data Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set
\$orderby	No	string	Field name to be used for sorting the result set in ascending order
\$filter	No	string	Filter condition to be applied; valid fields are businessPartner1ID, businessPartner2ID, and relationshipType
\$count	No	integer	Returns the number of entries in the database, with filter conditions taken into account (if applicable)

Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
businessPartner1ID, businessPartner2ID, relationshipType	eq, ne

## Request Example

```
/DeletedBPRElationships
```

Retrieves all deleted business partner relationships that are available in the database.

```
/DeletedBPRElationships?$top=25&$orderby='relationshipType'
```

Retrieves the first 25 deleted relationships. Data is retrieved in page mode to reduce network load and response times, and the retrieved deleted relationships are sorted by type.

```
/DeletedBPRElationships?$skip=10&$top=20
```

Retrieves 20 deleted relationships from the database where the first 10 deleted relationships are omitted. That is, the retrieved deleted relationships are those with sequence numbers 11 through 30.

## Response

### Response Status and Error Codes

Code	Reason
200	Deleted relationships retrieved successfully



## Related Information

[DeletedBPRelationship \[page 96\]](#)

### 2.4.5.3 Delete a DeletedBPRelationship

With this method, you physically delete a relationship with the specified ID.

Although "deleting a deleted relationship" may sound strange at first glance, this is exactly the purpose of this method. The DeletedBPRelationship concept represents a relationship between business partners that has played an active part in the system in the past, but has been logically deleted. As opposed to the logical deletion of a BPRelationship object, deleting a DeletedBPRelationship object means:

- **explicit** physical deletion of the DeletedBPRelationship object, and
- **implicit** physical deletion of the BPRelationship object that is represented by the corresponding DeletedBPRelationship object

In other words, deleting a DeletedBPRelationship object is the ultimate and irreversible way of removing all information about a particular relationship from the system.

#### Note

Be very careful before executing this method. Since it is the very purpose of the DeletedBPRelationship concept for you to be able to reconstruct the information stored for a relationship even after that relationship has been deleted from the system, we highly recommend that you make yourself aligned with all the relevant departments in your company before you delete a DeletedBPRelationship object.

## Request

**URI:** /DeletedBPRelationships('<relationshipID>')

**HTTP Method:** *DELETE*

**Permissions:** <bp>.pd

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token
If-Match	Yes	Check if cached version matches server version of the DeletedBPRelationship object

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
relationshipID	Yes	string	technical ID of the DeletedBPRElationship object	Path

## Request Example

```
/DeletedBPRElationships('9828317b1aee41e391aeb5cdd3b4fb6d')
```

## Response

### Response Status and Error Codes

Code	Reason
204	DeletedBPRElationship deleted successfully

## Related Information

[DeletedBPRElationship \[page 96\]](#)

## 2.5 BPRole

Describes a business partner's role within the Leonardo IoT platform.

This service lets you assign or unassign a predefined role to a business partner (person as well as organization). One business partner can have any number of roles assigned. The role categories that are currently available fall into two groups:

- Cloud platform roles describe the roles that are relevant for operating the cloud platform that provides Leonardo IoT services.
- Business partner roles describe the roles of the business partners from a business perspective.

The authorization needed for BPRoles is derived from the authorization of the involved business partner. This means that the user needs to have the write capability on the business partner to assign or unassign a role. This is because the business partner is changed with those requests. Also, to see the assignment, the read capability for the business partner is required.

### Base URI:

- Formal description: `https://<server address>[:<port number>][<path>]/BPRoles`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/BPRoles`

## Payload

For PUT and GET requests, only media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "roleID" : { "type": "string", "maxLength": 32 },
  "businessPartnerID" : { "type": "string", "minLength": 32, "maxLength": 32, "required": true },
  "roleCategory" : { "type": "string", "maxLength": 50, "required": true },
  "roleCategoryDescription": { "type": "string", "maxLength": 60, "readOnly": true }
}
```

### Field Explanation

Name	Description
roleID	Unique identifier of the role (automatically defined by the system)
businessPartnerID	Business partner to which the role is assigned
roleCategory	Coded type of the role. The following role categories are available: <ul style="list-style-type: none"><li>com.sap.appcore:CLD001</li><li>com.sap.appcore:CLD002</li><li>com.sap.appcore:CLD003</li><li>com.sap.appcore:BUP000</li><li>com.sap.appcore:BUP001</li><li>com.sap.appcore:BUP003</li></ul>
roleCategoryDescription	Semantic description of the role category <ul style="list-style-type: none"><li>Cloud Platform Provider</li><li>Cloud Platform Consumer</li><li>Cloud Platform Sub-Consumer</li><li>Vendor</li><li>Contact Person</li><li>Employee</li></ul>

## Methods

HTTP Method	Action	URI	Scopes
POST	<a href="#">Create a BPRole [page 104]</a>	/BPRoles	<bp>.c  <bp>.tenant.r

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read a BPRole [page 106]</a>	/BPRoles ('<roleID>')	<bp>.r <bp>.tenant.r
GET	<a href="#">Read all BPRoles [page 108]</a>	/BPRoles	<bp>.r <bp>.tenant.r
DELETE	<a href="#">Delete a BPRole [page 109]</a>	/BPRoles ('<roleID>')	<bp>.d <bp>.tenant.r

### i Note

There is no method available for modifying the values of an already existing role assignment. If changes are necessary, you can accomplish this by deleting the current role assignment and create it anew with the changed field values.

## Related Information

[Business Partner \[page 25\]](#)

[Organization \[page 28\]](#)

[Person \[page 55\]](#)

## 2.5.1 Create a BPRole

With this method, you create a new role assignment for a business partner

To create a new role assignment, you need to provide the ID of the business partner and the technical ID of the desired role category. This data must be provided via the respective payload fields of this method.

## Request

**URI:** /BPRoles

**HTTP Method:** *POST*

**Permissions:**

<bp>.c

<bp>.tenant.r

## Request Header Fields

Name	Required	Values
X-CSRF-Token	Yes	Cross site request forgery token

## Request Parameters

None.

## Request Example

```
/BPRoles
```

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "roleID" : { "type": "string", "maxLength": 32 },
  "businessPartnerID" : { "type": "string", "minLength": 32, "maxLength": 32, "required": true },
  "roleCategory" : { "type": "string", "maxLength": 50, "required": true },
  "roleCategoryDescription": { "type": "string", "maxLength": 60, "readOnly": true }
}
```

### Field Explanation

Name	Description
roleID	Unique identifier of the role (automatically defined by the system)
businessPartnerID	Business partner to which the role is assigned
roleCategory	Coded type of the role. The following role categories are available: <ul style="list-style-type: none"><li>com.sap.appcore:CLD001</li><li>com.sap.appcore:CLD002</li><li>com.sap.appcore:CLD003</li><li>com.sap.appcore:BUP000</li><li>com.sap.appcore:BUP001</li><li>com.sap.appcore:BUP003</li></ul>

Name	Description
roleCategoryDescription	Semantic description of the role category (inserted automatically by the system) <ul style="list-style-type: none"> <li>• Cloud Platform Provider</li> <li>• Cloud Platform Consumer</li> <li>• Cloud Platform Sub-Consumer</li> <li>• Vendor</li> <li>• Contact Person</li> <li>• Employee</li> </ul>

## Response

### Response Header Fields

Name	Description
Location	Path to the the newly created role

### Response Status and Error Codes

Code	Description
201	Role assigned successfully

## Related Information

[BPRole \[page 102\]](#)

## 2.5.2 Read a BPRole

Retrieves a specified business partner role assignment from the database

With this method, you send a request to the server to retrieve a particular BPRole object specified by its ID.

## Request

URI: `/BPRoles('<roleID>')`

HTTP Method: *GET*

Permissions:

<bp>.r

<bp>.tenant.r

## Query String Parameters

Name	Required	Data Type	Description	Parameter Type
roleID	Yes	string	Technical identifier of the role assignment	Path

## Request Example

```
/BPRoles('2CED0EAF62B4140A61F1404C2C5FBA9')
```

## Response

### Response Status and Error Codes

Code	Description
200	Role assignment retrieved successfully

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "roleID" : { "type": "string", "maxLength": 32 },
  "businessPartnerID" : { "type": "string", "minLength": 32, "maxLength": 32, "required": true },
  "roleCategory" : { "type": "string", "maxLength": 50, "required": true },
  "roleCategoryDescription": { "type": "string", "maxLength": 60, "readOnly": true }
}
```

## Related Information

[BPRole \[page 102\]](#)

## 2.5.3 Read all BPRoles

Retrieves a list of role assignments from the database

With this method, you send a request to the server to retrieve a subset of all BPRoles, according to the filter criteria provided. For a collective request for a set of roles, you have these options:

- You can restrict the set of roles to be retrieved by filter criteria based on the business partner involved. In fact, defining filter criteria is mandatory. The system does not accept an unrestricted request for a list of all business partner roles in the database.
- You can define that the set of matching roles shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

### i Note

Retrieving high numbers of objects from the database with only one collective GET request can lead to high system load and a significant increase of response times. Therefore, we highly recommend limiting the maximum number of returned objects to no more than 500 per method call. This can easily be accomplished with the help of the `$top` and `$skip` query parameters.

To avoid unwanted performance decline, collective GET requests are only supported up to a maximum of 500 objects that can be retrieved with one single call. Therefore, in use cases where it is crucial for you to ensure that all existing objects have been retrieved, you need to verify this by setting up a loop construct in your source code to repeat the request until the service method returns no further objects.

## Request

**URI:** `/BPRoles`

**HTTP Method:** `GET`

**Permissions:**

`<bp>.r`

`<bp>.tenant.r`

### Query String Parameters

Name	Required	Data Type	Description
<code>\$skip</code>	No	integer	Number of the first n records to be excluded from the result set
<code>\$top</code>	No	integer	Number of records to include in the result set
<code>\$filter</code>	Yes	string	Filter condition to be applied; valid field: <code>businessPartnerID</code>
<code>\$count</code>	No	integer	Returns the number of entries in the database, with filter conditions taken into account (if applicable)



#### Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
businessPartnerID	eq, ne

### Request Example

```
/BPRoles
```

Retrieves all roles that are available in the database. Note that this unrestricted request is not supported due to unacceptable system load.

```
/BPRoles?$top=5&$filter=businessPartnerID eq 'd45add8e7f343588e4a504a3a2fcc70'
```

Retrieves the first 5 role assignments for the specified business partner.

```
/BPRoles?$filter=businessPartnerID eq 'd45add8e7f343588e4a504a3a2fcc70'
```

Retrieves all role assignments for business partner with ID 'd45add8e7f343588e4a504a3a2fcc70'.

## Response

### Response Status and Error Codes

Code	Description
200	Role assignments retrieved successfully

## Related Information

[BPRole \[page 102\]](#)

## 2.5.4 Delete a BPRole

With this method, you delete a BPRole object with the specified ID.

From a business perspective, it is normally not desired to have a business partner without role information in the system. Therefore, we recommend ensuring that each business partner has at least one role assigned.

## Request

**URI:** `/BPRoles('<roleID>')`

**HTTP Method:** `DELETE`

#### Permissions:

<bp>.d

<bp>.tenant.r

#### Request Header Fields

Name	Required	Values
X-CSRF-Token	Yes	Cross site request forgery token

#### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
roleID	Yes	string	Technical identifier of the role	Path

#### Request Example

```
/BPRoles('2CED0EAF62B4140A61F1404C2C5FBA9')
```

## Response

#### Response Status and Error Codes

Code	Description
204	Role assignment deleted successfully

## Related Information

[BPRole \[page 102\]](#)

[DeletedBPRole \[page 110\]](#)

## 2.5.5 DeletedBPRole

A DeletedBPRole object is a physical backup for a logically deleted DeletedBPRole object.

Whenever a user deletes a business partner role, this deletion is a logical rather than a physical deletion. That is, from a user's perspective, the deleted role is no longer available in the system and cannot be accessed anymore. However, in the background, the system takes several steps to ensure that the deleted role can still be restored, if need be. Here are some use cases where access to a previously deleted role may be required:

- Restoring a role that has accidentally been deleted
- Retrieving a role for auditing purposes

- Retrieving a role in the context of a lawsuit

Due to the special nature of the `DeletedBPRole` object, access to objects of this kind is limited to system administrators with special authorizations. Moreover, a `DeletedBPRole` object has a restricted set of methods that can be accessed via the Leonardo IoT API. For example, there is no `Create` method available, nor can it be modified. This is to make sure that a `DeletedBPRole` object can only be created in the system according to certain predefined and hard-coded rules. These rules enforce that `DeletedBPRole` objects can only exist under well-defined circumstances laid down in legal frameworks that are relevant for operating an IoT scenario.

Write access to `DeletedBPRole` objects residing in a tenant other than the one to which a user is logged on is only allowed for users who are acting in the role of a cloud platform provider.

#### Base URI:

- Formal description: `https://<server address>[:<port number>]/<path>/DeletedBPRoles`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/DeletedBPRoles`

**Permissions:** `<bp>.pd`

## Methods

HTTP Method	Action	URI	Scopes
<i>GET</i>	<a href="#">Read a DeletedBPRole [page 113]</a>	<code>/DeletedBPRoles('&lt;roleID&gt;')</code>	<code>&lt;bp&gt;.pd</code>
<i>GET</i>	<a href="#">Read all DeletedBPRoles [page 114]</a>	<code>/DeletedBPRoles</code>	<code>&lt;bp&gt;.pd</code>
<i>DELETE</i>	<a href="#">Delete a DeletedBPRole [page 115]</a>	<code>/DeletedBPRoles('&lt;roleID&gt;')</code>	<code>&lt;bp&gt;.pd</code>

### i Note

As already mentioned above, the `DeletedBPRole` object has neither a `POST` nor a `PUT` method:

- There is no way of explicitly creating a `DeletedBPRole` object by a method call via the interface. Instead, a `DeletedBPRole` object is created automatically by the system whenever a `BPRole` object is deleted.
- Once the system has created a `DeletedBPRole` object, there is no way of modifying its field values, neither explicitly nor implicitly. This is due to the strict legal requirements that are applicable in the use cases outlined above.

## Payload

For PUT and GET requests, only media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "roleID" : { "type": "string", "maxLength": 32 },
  "businessPartnerID" : { "type": "string", "minLength": 32, "maxLength": 32, "required": true },
  "roleCategory" : { "type": "string", "maxLength": 50, "required": true },
  "roleCategoryDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
}
```

### Field Explanation

Name	Description
roleID	Unique identifier of the role (automatically defined by the system)
businessPartnerID	Business partner to which the role is assigned
roleCategory	Coded type of the role. The following role categories are available: <ul style="list-style-type: none"><li>com.sap.appcore:CLD001</li><li>com.sap.appcore:CLD002</li><li>com.sap.appcore:CLD003</li><li>com.sap.appcore:BUP000</li><li>com.sap.appcore:BUP001</li><li>com.sap.appcore:BUP003</li></ul>
roleCategoryDescription	Semantic description of the role category <ul style="list-style-type: none"><li>Cloud Platform Provider</li><li>Cloud Platform Consumer</li><li>Cloud Platform Sub-Consumer</li><li>Vendor</li><li>Contact Person</li><li>Employee</li></ul>

## Related Information

[BPRole \[page 102\]](#)

[Delete a BPRole \[page 109\]](#)

## 2.5.5.1 Read a DeletedBPRole

Retrieves the specified DeletedBPRole object from the database

With this method, you send a request to the server to retrieve a particular DeletedBPRole object specified by its ID.

If the DeletedBPRole object is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

### Request

**URI:** /DeletedBPRoles('<roleID>')

**HTTP Method:** *GET*

**Permissions:** <bp>.pd

### Request Header Fields

Name	Required	Description
If-Match	No	Check if cached version matches server version of the role

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
roleID	Yes	string	Technical ID of the DeletedBPRole object	Path

### Request Example

```
/DeletedBPRoles('a325e746f0e749f38d3a451525a09a13')
```

### Response

#### Format

**Format:** *JSON*

### Response Status and Error Codes

Code	Reason
200	DeletedBPRole retrieved successfully

## Related Information

[DeletedBPRole \[page 110\]](#)

### 2.5.5.2 Read all DeletedBPRoles

Retrieves a list of deleted business partner roles from the database

With this method, you send a request to the server to retrieve a subset of all deleted business partner roles, according to the filter criteria provided. For a collective request for a set of deleted roles, you have these options:

- You can define that the set of matching deleted roles shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

## Request

**URI:** /DeletedBPRoles

**HTTP Method:** *GET*

**Permissions:** <bp>.pd

## Query String Parameters

Name	Required	Data Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set
\$orderby	No	string	Field name to be used for sorting the result set in ascending order
\$filter	Yes	string	Filter condition to be applied; valid field: businessPartnerID
\$count	No	integer	Returns the number of entries in the database, with filter conditions taken into account (if applicable)

Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
businessPartnerID	eq, ne

## Request Example

```
/DeletedBPRoles
```

Retrieves all deleted business partner roles that are available in the database.

```
/DeletedBPRoles?$top=25&$orderby='roleID'
```

Retrieves the first 25 deleted roles. Data is retrieved in page mode to reduce network load and response times, and the retrieved deleted roles are sorted by their ID.

```
/DeletedBPRoles?$skip=10&$top=20
```

Retrieves 20 deleted roles from the database where the first 10 deleted roles are omitted. That is, the retrieved deleted roles are those with sequence numbers 11 through 30.

## Response

### Response Status and Error Codes

Code	Reason
200	Deleted roles retrieved successfully

## Related Information

[DeletedBPRole \[page 110\]](#)

### 2.5.5.3 Delete a DeletedBPRole

With this method, you physically delete a role with the specified ID.

Although "deleting a deleted role" may sound strange at first glance, this is exactly the purpose of this method. The DeletedBPRole concept represents a business partner role that has played an active part in the system in the past, but has been logically deleted. As opposed to the logical deletion of a BPRole object, deleting a DeletedBPRole object means:

- **explicit** physical deletion of the DeletedBPRole object, and
- **implicit** physical deletion of the BPRole object that is represented by the corresponding DeletedBPRole object

In other words, deleting a DeletedBPRole object is the ultimate and irreversible way of removing all information about a particular role from the system.

#### i Note

Be very careful before executing this method. Since it is the very purpose of the DeletedBPRole concept for you to be able to reconstruct the information stored for a role even after that role has been deleted from the system, we highly recommend that you make yourself aligned with all the relevant departments in your company before you delete a DeletedBPRole object.

## Request

URI: /DeletedBPRoles('<roleID>')

HTTP Method: *DELETE*

Permissions: <bp>.pd

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token
If-Match	Yes	Check if cached version matches server version of the DeletedBPRole object

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
roleID	Yes	string	technical ID of the DeletedBPRole object	Path

### Request Example

```
/DeletedBPRoles('a325e746f0e749f38d3a451525a09a13')
```

## Response

### Response Status and Error Codes

Code	Reason
204	DeletedBPRole deleted successfully

## Related Information

[DeletedBPRole \[page 110\]](#)

## 2.6 BPUserRelationship

Describes the relationship of a user to a business partner

A BPUserRelationship object describes the relationship between a user and a business partner. While the concept of the business partner deals with the business-oriented perspective on an actor in a particular



business scenario, the user is a technical entity that is needed for a business partner to actually log on to the SAP Leonardo IoT platform. Although the business partner concept comprises both organizations as well as persons, a user is normally associated only with business partners of type person.

To fully define the relationship between a user and a business partner, the user ID and the business partner ID must be provided. The technical ID of the relationship itself is automatically calculated and assigned by the system upon creation.

#### Base URI:

- Formal description: `https://<server address>[:<port number>][<path>]/BPUserRelationships`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/BPUserRelationships`

## Payload

For POST and GET, only media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "relationshipID" : { "type": "string", "maxLength": 32 },
  "businessPartnerID" : { "type": "string", "minLength": 32, "maxLength": 32,
    "required": true },
  "userNameID" : { "type": "string", "minLength": 1, "maxLength": 255,
    "required": true }
}
```

#### Field Explanation

Name	Description
relationshipID	Unique identifier of the relationship (automatically defined by the system)
businessPartnerID	Business partner to which the user is related
userNameID	User to which the business partner is related

## Methods

HTTP Method	Action	URI	Scope
POST	<a href="#">Create a BPUserRelationship [page 118]</a>	/BPUserRelationships	<bp>.c  <bp>.tenant.r
GET	<a href="#">Read a BPUserRelationship [page 120]</a>	/BPUserRelationships('<relationshipID>')	<bp>.r  <bp>.tenant.r

HTTP Method	Action	URI	Scope
<i>GET</i>	<a href="#">Read all BPUserRelationships [page 121]</a>	/BPUserRelationships	<bp>.r <bp>.tenant.r
<i>GET</i>	<a href="#">Read all Matching BPUserRelationships [page 123]</a>	/BPUserRelationships	<bp>.r <bp>.tenant.r
<i>DELETE</i>	<a href="#">Delete a BPUserRelationship [page 125]</a>	/BPUserRelationships('<relationshipID>')	<bp>.d <bp>.tenant.r

### Note

There is no method available for modifying the values of an already existing relationship between a business partner and a user. If changes are necessary, you can accomplish this by deleting the current relationship and create it anew with the changed field values.

## Related Information

[Business Partner \[page 25\]](#)

[Person \[page 55\]](#)

## 2.6.1 Create a BPUserRelationship

With this method, you create a new relationship between business partner and a user

To create a new relationship, you need to provide the IDs of the business partner and the user. This data must be provided via the respective payload fields of this method.

## Request

**URI:** /BPUserRelationships

**HTTP Method:** *POST*

**Permissions:**

<bp>.c

<bp>.tenant.r

## Request Header Fields

Name	Required	Values
X-CSRF-Token	Yes	Cross site request forgery token

## Request Parameters

None.

## Request Example

```
/BPUserRelationships
```

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "relationshipID" : { "type": "string", "maxLength": 32 },
  "businessPartnerID" : { "type": "string", "minLength": 32, "maxLength": 32,
    "required": true },
  "userNameID" : { "type": "string", "minLength": 1, "maxLength": 255,
    "required": true }
}
```

## Response

### Response Header Fields

Name	Description
Location	Path to the the newly created relationship

### Response Status and Error Codes

Code	Description
201	Relationship created successfully

## Related Information

[BPUserRelationship \[page 116\]](#)

## 2.6.2 Read a BPUserRelationship

Retrieves a specified relationship from the database

With this method, you send a request to the server to retrieve a particular BPUserRelationship object specified by its ID.

### Request

**URI:** /BPUserRelationships('<relationshipID>')

**HTTP Method:** *GET*

**Permissions:**

<bp>.r

<bp>.tenant.r

### Query String Parameters

Name	Required	Data Type	Description	Parameter Type
relationshipID	Yes	string	Technical identifier of the relationship	Path

### Request Example

```
/BPUserRelationship('2CED0EAFC62B4140A61F1404C2C5FBA9')
```

### Response

#### Response Status and Error Codes

Code	Description
200	Relationship retrieved successfully

### Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "relationshipID" : { "type": "string", "maxLength": 32 },
  "businessPartnerID" : { "type": "string", "minLength": 32, "maxLength": 32,
    "required": true },
  "userID" : { "type": "string", "minLength": 1, "maxLength": 255,
    "required": true }
```

```
}
```

## Related Information

[BPUserRelationship](#) [page 116]

### 2.6.3 Read all BPUserRelationships

Retrieves a list of relationships between users and business partners from the database

With this method, you send a request to the server to retrieve a subset of all BPUserRelationships, according to the filter criteria provided. For a collective request for a set of user relationships, you have these options:

- You can restrict the set of relationships to be retrieved by filter criteria based on the business partner or the user.
- You can define that the set of matching relationships shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

#### Note

Retrieving high numbers of objects from the database with only one collective GET request can lead to high system load and a significant increase of response times. Therefore, we highly recommend limiting the maximum number of returned objects to no more than 500 per method call. This can easily be accomplished with the help of the `$top` and `$skip` query parameters.

To avoid unwanted performance decline, collective GET requests are only supported up to a maximum of 500 objects that can be retrieved with one single call. Therefore, in use cases where it is crucial for you to ensure that all existing objects have been retrieved, you need to verify this by setting up a loop construct in your source code to repeat the request until the service method returns no further objects.

## Request

**URI:** `/BPUserRelationships`

**HTTP Method:** `GET`

**Permissions:**

`<bp>.r`

`<bp>.tenant.r`

## Query String Parameters

Name	Required	Data Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set
\$filter	No	string	Filter condition to be applied; valid fields are <code>businessPartnerID</code> and <code>userNameID</code>
\$count	No	integer	Returns the number of entries in the database, with filter conditions taken into account (if applicable)

### Note

Although from a technical perspective all the query parameters listed above are **not** mandatory, we still recommend restricting the result set of elements.

Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
<code>businessPartnerID</code> , <code>userNameID</code>	<code>eq</code> , <code>ne</code>

## Request Example

```
/BPUserRelationships
```

Retrieves all relationships that are available in the database.

```
/BPUserRelationships?$top=25
```

Retrieves the first 25 relationships. Data is retrieved in page mode to reduce network load and response times.

```
/BPUserRelationships?$filter=businessPartnerID eq  
'8ab31cdfel1a44ee0a33cc2cae9d2c27f'
```

Retrieves all relationships for business partner `8ab31cdfel1a44ee0a33cc2cae9d2c27f`.

## Response

### Response Status and Error Codes

Code	Description
200	Relationships retrieved successfully

## Related Information

[BPUserRelationship](#) [page 116]

### 2.6.4 Read all Matching BPUserRelationships

Retrieves a list of BPUserRelationship objects matching the filter criteria provided

With this method, you send a request to the server to retrieve a subset of all relationships between business partners and users, according to the filter criteria provided. This method offers an advanced filter functionality that goes beyond the standard filters used for collective requests of most of the other services available in the SAP Leonardo IoT platform. Here, you can choose between two different approaches to define a filter criterion:

- A comma-separated list of individual person IDs
- A comma-separated list of individual user names

The method then returns a collection of data sets for each element specified in the request. This method is especially useful for all kinds of administrative tasks dealing with the relationship between onboarded persons and users. For example, you could set up a list of persons and find out which of these persons has no user assigned yet.

## Request

**URI:** /BPUserRelationships

**HTTP Method:** [GET](#)

**Permissions:**

<bp>.r

<bp>.tenant.r

## Query String Parameters

Parameter	Required	Data Type	Description
listFilterID	No	string	comma-separated list of IDs referring to the businessPartnerID attribute of a Person
listFilterUserName	No	string	comma-separated list of IDs referring to the userNameID attribute of a BPUserRelationship

## Request Example

```
/BPUserRelationships?  
listFilterID=adbb2c0fdb0a4b9e99327a52c41fc720,adbb2c0fdb0a4b9e99327a52c41fc721,ad  
bb2c0fdb0a4b9e99327a52c41fc722
```

Retrieves three `BPUserRelationship` objects with the specified person IDs from the database. This lets you determine the user names that have been assigned to these persons.

```
/BPUserRelationships?listFilterUserName=user01,user02,user47
```

Retrieves three `BPUserRelationship` objects with the specified user names from the database. This lets you determine the persons to which these users have been assigned.

## Response

### Response Status and Error Codes

Code	Description
200	List of relationships retrieved successfully
400	Bad request (for example, invalid filter)
403	Forbidden (user not authorized to access relationships)

## Payload

Format: *JSON*

Only media type JSON is supported. The JSON for this method is an array of multiple occurrences of the following structure (one for each retrieved relationship):

```
{
  "relationshipID"      : { "type": "string", "maxLength": 32 },
  "businessPartnerID"  : { "type": "string", "minLength": 32, "maxLength": 32,
    "required": true },
  "userNameID"         : { "type": "string", "minLength": 1, "maxLength": 255,
    "required": true }
}
```

## Related Information

[BPUserRelationship \[page 116\]](#)



## 2.6.5 Delete a BPUserRelationship

With this method, you disconnect a business partner from its user

As a consequence of disconnecting a business partner from its user, that business partner can no longer access objects for which he or she has been granted any access rights based on the UserGroup to which the user belongs.

### Request

**URI:** /BPUserRelationships('<relationshipID>')

**HTTP Method:** *DELETE*

**Permissions:**

<bp>.d

<bp>.tenant.r

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross site request forgery token

### Query String Parameters

Name	Required	Data Type	Description	Parameter Type
relationshipID	Yes	string	Technical identifier of the relationship	Path

### Request Example

```
/BPUserRelationship('2CED0EAF62B4140A61F1404C2C5FBA9')
```

### Response

#### Response Status and Error Codes

Code	Description
204	Relationship deleted successfully

## Related Information

[BPUserRelationship \[page 116\]](#)

[DeletedBPUserRelationship \[page 126\]](#)

## 2.6.6 DeletedBPUserRelationship

A `DeletedBPUserRelationship` object is a physical backup for a logically deleted `BPUserRelationship` object.

Whenever a user deletes a relationship between a business partner and a user object as defined by the identity provider service from an IoT scenario, this deletion is a logical rather than a physical deletion. That is, from a user's perspective, the deleted relationship is no longer available in the system and cannot be accessed anymore. However, in the background, the system takes several steps to ensure that the deleted relationship can still be restored, if need be. Here are some use cases where access to a previously deleted relationship may be required:

- Restoring a relationship that has accidentally been deleted
- Retrieving a relationship for auditing purposes
- Retrieving a relationship in the context of a lawsuit

Due to the special nature of the `DeletedBPUserRelationship` object, access to objects of this kind is limited to system administrators with special authorizations. Moreover, a `DeletedBPUserRelationship` object has a restricted set of methods that can be accessed via the Leonardo IoT API. For example, there is no `Create` method available, nor can it be modified. This is to make sure that a `DeletedBPUserRelationship` objects can only be created in the system according to certain predefined and hard-coded rules. These rules enforce that `DeletedBPUserRelationship` objects can only exist under well-defined circumstances laid down in legal frameworks that are relevant for operating an IoT scenario.

Write access to `DeletedBPUserRelationship` objects residing in a tenant other than the one to which a user is logged on is only allowed for users who are acting in the role of a cloud platform provider.

### Base URI:

- Formal description: `https://<server address>[:<port number>]/[<path>]/DeletedBPUserRelationships`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/DeletedBPUserRelationships`

**Permissions:** `<bp>.pd`

## Methods

HTTP Method	Action	URI	Scopes
GET	[Link to the method topic] <a href="#">Read a DeletedBPUserRelationship</a> [page 128]	/DeletedBPUserRelationships ('<relationshipID>')	<bp>.pd
GET	[Link to the method topic] <a href="#">Read all DeletedBPUserRelationships</a> [page 129]	/DeletedBPUserRelationships ()	<bp>.pd
DELETE	<a href="#">Delete a DeletedBPUserRelationship</a> [page 130]	/DeletedBPUserRelationships ('<relationshipID>')	<bp>.pd

### i Note

As already mentioned above, the DeletedBPUserRelationship object has neither a POST nor a PUT method:

- There is no way of explicitly creating a DeletedBPUserRelationship object by a method call via the interface. Instead, a DeletedBPUserRelationship object is created automatically by the system whenever a BPUserRelationship object is deleted.
- Once the system has created a DeletedBPUserRelationship object, there is no way of modifying its field values, neither explicitly nor implicitly. This is due to the strict legal requirements that are applicable in the use cases outlined above.

## Payload

Only media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "relationshipID" : { "type": "string", "maxLength": 32 },
  "businessPartnerID" : { "type": "string", "minLength": 32, "maxLength": 32,
    "required": true },
  "userNameID" : { "type": "string", "minLength": 1, "maxLength": 255,
    "required": true }
}
```

Field Explanation

Name	Description
relationshipID	Unique identifier of the relationship (automatically defined by the system)
businessPartnerID	Business partner to which the user is related
userNameID	User to which the business partner is related

## Related Information

[BPUserRelationship \[page 116\]](#)

[Delete a BPUserRelationship \[page 125\]](#)

### 2.6.6.1 Read a DeletedBPUserRelationship

Retrieves the specified DeletedBPUserRelationship object from the database

With this method, you send a request to the server to retrieve a particular DeletedBPUserRelationship object specified by its ID.

If the DeletedBPUserRelationship object is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

## Request

**URI:** `/DeletedBPUserRelationships('<relationshipID>')`

**HTTP Method:** [GET](#)

**Permissions:** `<bp>.pd`

## Request Header Fields

Name	Required	Description
If-Match	No	Check if cached version matches server version of the relationship

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
relationshipID	Yes	string	Technical ID of the DeletedBPUser-Relationship object	Path

## Request Example

```
/DeletedBPUserRelationships('b474687b75de4afa97d7bc92f79c3531')
```

## Response

### Format

**Format:** [JSON](#)

## Response Status and Error Codes

Code	Reason
200	DeletedBPUserRelationship retrieved successfully

## Related Information

[DeletedBPUserRelationship \[page 126\]](#)

## 2.6.6.2 Read all DeletedBPUserRelationships

Retrieves a list of deleted relationships between business partners and system users from the database

With this method, you send a request to the server to retrieve a subset of all deleted relationships between business partners (persons or organizations) and system users provided by the identity provider service, according to the filter criteria provided. For a collective request for a set of deleted relationships, you have these options:

- You can define that the set of matching deleted relationships shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

## Request

**URI:** /DeletedBPUserRelationships

**HTTP Method:** [GET](#)

**Permissions:** <bp>.pd

## Query String Parameters

Name	Required	Data Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set
\$orderby	No	string	Field name to be used for sorting the result set in ascending order
\$filter	No	string	Filter condition to be applied; valid fields are businessPartnerID and userNameID
\$count	No	integer	Returns the number of entries in the database, with filter conditions taken into account (if applicable)

Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
businessPartnerID, userNameID	eq, ne

## Request Example

```
/DeletedBPUserRelationships
```

Retrieves all deleted relationships between business partner and users that are available in the database.

```
/DeletedBPUserRelationships?$top=25&$orderby='userID'
```

Retrieves the first 25 deleted relationships. Data is retrieved in page mode to reduce network load and response times, and the retrieved deleted relationships are sorted by the ID of the identity provider user.

```
/DeletedBPUserRelationships?$skip=10&$top=20
```

Retrieves 20 deleted relationships from the database where the first 10 deleted relationships are omitted. That is, the retrieved deleted relationships are those with sequence numbers 11 through 30.

## Response

### Response Status and Error Codes

Code	Reason
200	Deleted relationships retrieved successfully

## Related Information

[DeletedBPUserRelationship](#) [page 126]

### 2.6.6.3 Delete a DeletedBPUserRelationship

With this method, you physically delete a relationship with the specified ID.

Although "deleting a deleted relationship" may sound strange at first glance, this is exactly the purpose of this method. The DeletedBPUserRelationship concept represents a relationship between business partners that has played an active part in the system in the past, but has been logically deleted. As opposed to the logical deletion of a BPUserRelationship object, deleting a DeletedBPUserRelationship object means:

- **explicit** physical deletion of the DeletedBPUserRelationship object, and
- **implicit** physical deletion of the BPUserRelationship object that is represented by the corresponding DeletedBPUserRelationship object

In other words, deleting a DeletedBPUserRelationship object is the ultimate and irreversible way of removing all information about a particular relationship from the system.

### Note

Be very careful before executing this method. Since it is the very purpose of the DeletedBPUserRelationship concept for you to be able to reconstruct the information stored for a relationship even after that relationship has been deleted from the system, we highly recommend that you make yourself aligned with all the relevant departments in your company before you delete a DeletedBPUserRelationship object.

## Request

**URI:** /DeletedBPUserRelationships('<relationshipID>')

**HTTP Method:** *DELETE*

**Permissions:** <bp>.pd

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token
If-Match	Yes	Check if cached version matches server version of the DeletedBPUserRelationship object

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
relationshipID	Yes	string	technical ID of the DeletedBPUserRelationship object	Path

### Request Example

```
/DeletedBPUserRelationships('b474687b75de4afa97d7bc92f79c3531')
```

## Response

### Response Status and Error Codes

Code	Reason
204	DeletedBPUserRelationship deleted successfully

## Related Information

[DeletedBPUserRelationship \[page 126\]](#)



## 3 Location

### Cross-functional services for location data

The services described in this section serve the purpose of providing a self-contained instrument for managing location-related data. These services can hold both standard address data as well as geographical coordinates in their payload.

Location services are used within the SAP Leonardo IoT platform for things. Here, the built-in support for geographical coordinates is especially useful because things are an extremely generic concept and can represent almost everything. With that, it is important not to rely on the assumed identity of a location with a postal address. In fact, many things may exist without such an address. For example, if a thing represents a building or a technical installation equipped with sensors that is located outside of developed areas, using geocoordinates instead of address data can often be the only possible way of describing the location of that building. This may be the case for bridges, power plants, wind turbines, offshore drilling platforms, and so on.

The decision to model location services in a self-contained application within the SAP Leonardo IoT platform is motivated by the idea of providing a uniform and consistent way of handling location data that can easily be accessed by all domain-specific applications running in the same environment. This should help increase reuse effects and prevent you from reinventing the wheel.

#### i Note

The business partner services of SAP Leonardo IoT (that is, Person and Organization) do **not** make use of this service. Instead, they come with their own location model, which is implemented in the `communicationData` block of the service payload. However, you are of course free to implement a connection between the Location service and the business partner services manually. Such a connection can be useful when you have to onboard a high number of persons or organizations at the same location and want to ensure data consistency. Moreover, the payload fields that both the Location service and the `communicationData` block have in common are modeled in the same way with respect to types and field lengths. With that, you can easily take over the field contents from a uniform Location object into the corresponding fields of individual business partner entities.

As of November 2019, the legacy [Location \[page 134\]](#) service has been declared deprecated. For new projects, use the Geolocation service instead. For more information, see [Geolocations Overview \[page 151\]](#). To migrate existing projects from the old Location service to the new Geolocation service, see the [Migration Guide](#).

### Related Information

[Location \[page 134\]](#)

[DeletedLocation \[page 145\]](#)

[Geolocations Overview \[page 151\]](#)

## 3.1 Location

Contains the geographical position of a thing

### i Note

The Location service is deprecated as of version 1911b (November 2019) of SAP Leonardo IoT. It will be dismantled in the near future. The old service is replaced by the new [Geolocation \[page 151\]](#) service. Further usage of the Location service is discouraged. We strongly recommend checking your source code and replacing any existing usages of the Location service by references to the new Geolocation service.

A Location object contains the postal address and the geolocation data in coordinate system 4326 (standard GPS data). It is used to describe the geographical position of a thing that is registered on the Leonardo IoT platform. The set of fields in the payload is similar to the communication data field block of a business partner, but without phone numbers or e-mail address.

In addition to the usual address fields, it is also possible to specify an absolute geographical position with the help of the `longitude` and `latitude` fields. This is useful when you need to record the position of a thing that is located at a place without postal address. For example, this can be the case for technical installations like a wind turbine or a vibration sensor installed on a railroad bridge.

Location objects are **not** secured via instance authorization, but they are tenant isolated. This means that read access, creation, update, and deletion actions are only possible within the tenant to which a user is logged on. Access to locations of other tenants is not possible. Even a user acting as a Cloud Platform Provider (TO) cannot access locations of different tenants, because access is **not** granted via capabilities. As a consequence, there is no way of inheriting access rights from the root object group.

### i Note

To assign a location to a thing, you enter the `locationID` value of the desired location into the `_location` field of a thing.

### Base URI:

- Formal description: `https://<server address>[:<port number>][<path>]/Locations`
- Example for a base URI in a cloud foundry environment: `https://location.cfapps.eu10.hana.ondemand.com/Locations`

## Methods

HTTP Method	Action	URI	Scopes
POST	<a href="#">Create a Location [page 136]</a>	/Locations	<loc>.c
GET	<a href="#">Read a Location [page 137]</a>	/Locations('<locationID>')	<loc>.r
GET	<a href="#">Read all Locations [page 138]</a>	/Locations	<loc>.r

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read all Matching Locations [page 140]</a>	/Locations	<loc>.r
PUT	<a href="#">Update a Location [page 142]</a>	/Locations ('<locationID>')	<loc>.u
DELETE	<a href="#">Delete a Location [page 144]</a>	/Locations ('<locationID>')	<loc>.d

## Payload

For PUT, POST and GET, only media type JSON is supported. The JSON for these methods has the following structure:

```
{ "basicData": {
  "locationID"      : { "type": "string", "maxLength": 32 },
  "tenant"          : { "type": "string", "maxLength": 36,
"required": true },
  "etag"            : { "type": "string" }
  "locationData": {
    "streetName"     : { "type": "string", "maxLength": 255 },
    "houseNumber"    : { "type": "string", "maxLength": 10 },
    "cityName"        : { "type": "string", "maxLength": 255 },
    "district"        : { "type": "string", "maxLength": 255 },
    "postalCode"      : { "type": "string", "maxLength": 10 },
    "country"         : { "type": "string", "maxLength": 2 },
    "countryDescription" : { "type": "string", "maxLength": 60,
"readOnly": true },
    "region"          : { "type": "string", "maxLength": 3 },
    "regionDescription" : { "type": "string", "maxLength": 60,
"readOnly": true },
    "longitude"       : { "type": "number"},
    "latitude"        : { "type": "number"}
  }
}
```

## Related Information

[Thing \[page 705\]](#)

## 3.1.1 Create a Location

With this method, you create a new Location object

### Request

URI: `/Locations`

HTTP Method: *POST*

Permissions: `<loc>.c`

### Request Header Fields

Name	Required	Values
X-CSRF-Token	Yes	Cross site request forgery token

### Request Parameters

None.

### Request Example

```
/Locations
```

### Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{ "basicData": {
  "locationID"      : { "type": "string", "maxLength": 32 },
  "tenant"          : { "type": "string", "maxLength": 36,
"required": true },
  "etag"            : { "type": "string" }
}
  "locationData": {
    "streetName"      : { "type": "string", "maxLength": 255 },
    "houseNumber"     : { "type": "string", "maxLength": 10 },
    "cityName"        : { "type": "string", "maxLength": 255 },
    "district"        : { "type": "string", "maxLength": 255 },
    "postalCode"      : { "type": "string", "maxLength": 10 },
    "country"         : { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60,
"readOnly": true },
    "region"          : { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60,
"readOnly": true },
    "longitude"       : { "type": "number"},
    "latitude"        : { "type": "number"}
  }
}
```

## Response

### Response Headers

Name	Description
Location	Path to the the newly created Location object
Etag	Identifies the newly created Location instance

### Response Status and Error Codes

Code	Description
201	Location created successfully

## Related Information

[Location \[page 134\]](#)

## 3.1.2 Read a Location

Retrieves the location with the specified ID from the database

With this method, you send a request to the server to retrieve a particular Location object specified by its ID.

## Request

URI: `/Locations('<locationID>')`

HTTP Method: [GET](#)

Permissions: `<loc>.r`

### Query String Parameters

Name	Required	Data Type	Description	Parameter Type
locationID	Yes	string	Unique identifier of the Location object	Path

### Request Example

```
/Locations('4BB5CE97334A44DC850CC35062AE2B26')
```

## Response

### Response Status and Error Codes

Code	Description
200	Location retrieved successfully

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{ "basicData": {
  "locationID"      : { "type": "string", "maxLength": 32 },
  "tenant"          : { "type": "string", "maxLength": 36,
"required": true },
  "etag"            : { "type": "string" }
}
"locationData": {
  "streetName"      : { "type": "string", "maxLength": 255 },
  "houseNumber"     : { "type": "string", "maxLength": 10 },
  "cityName"        : { "type": "string", "maxLength": 255 },
  "district"        : { "type": "string", "maxLength": 255 },
  "postalCode"      : { "type": "string", "maxLength": 10 },
  "country"         : { "type": "string", "maxLength": 2 },
  "countryDescription" : { "type": "string", "maxLength": 60,
"readOnly": true },
  "region"          : { "type": "string", "maxLength": 3 },
  "regionDescription" : { "type": "string", "maxLength": 60,
"readOnly": true },
  "longitude"       : { "type": "number"},
  "latitude"        : { "type": "number"}
}
}
```

## Related Information

[Location \[page 134\]](#)

### 3.1.3 Read all Locations

Retrieves a list of locations from the database

With this method, you send a request to the server to retrieve a subset of all Location objects, according to the filter criteria provided. For a collective request for a set of locations, you have these options:

- You can restrict the set of locations to be retrieved by filter criteria based on the postal code, region, or country. In fact, defining filter criteria is mandatory. The system does not accept an unrestricted request for a list of all locations in the database.

- You can define that the set of matching locations shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

### i Note

Retrieving high numbers of objects from the database with only one collective GET request can lead to high system load and a significant increase of response times. Therefore, we highly recommend limiting the maximum number of returned objects to no more than 500 per method call. This can easily be accomplished with the help of the `$top` and `$skip` query parameters.

To avoid unwanted performance decline, collective GET requests are only supported up to a maximum of 500 objects that can be retrieved with one single call. Therefore, in use cases where it is crucial for you to ensure that all existing objects have been retrieved, you need to verify this by setting up a loop construct in your source code to repeat the request until the service method returns no further objects.

## Request

**URI:** `/Locations`

**HTTP Method:** `GET`

**Permissions:** `<loc>.r`

### Query String Parameters

Name	Required	Data Type	Description
<code>\$skip</code>	No	integer	Number of the first n records to be excluded from the result set
<code>\$top</code>	No	integer	Number of records to include in the result set
<code>\$filter</code>	No	string	Filter condition to be applied; valid fields are <code>country</code> , <code>region</code> , <code>postalCode</code> , <code>longitude</code> , and <code>latitude</code> .

### i Note

Although from a technical perspective all the query parameters listed above are **not** mandatory, the requirement of restricting the result set of elements still remains in effect. It is only up to you to define a proper combination of parameters to fulfill this requirement.

Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
longitude, latitude	eq, ne, lt, le, gt, ge
postalCode, country, region	eq, ne

## Request Example

```
/Locations
```

Retrieves all locations that are available in the database. Note that this unrestricted request is not supported due to unacceptable system load.

```
/Locations?$top=25
```

Retrieves the first 25 locations. Data is retrieved in page mode to reduce network load and response times.

```
/Locations?$filter=country eq IN
```

Retrieves all locations from the database that indicate a place in India.

```
/Locations?$filter=longitude gt 5.8 and longitude lt 11 and latitude lt 55 and  
latitude gt 51
```

Retrieves all locations from the database within a rectangle that roughly surrounds the Republic of Ireland.

```
/Locations?$filter=country eq DE and postalCode eq 69190
```

Retrieves all locations from the database that belong to the city of Walldorf in Germany.

## Response

### Response Status and Error Codes

Code	Description
200	Locations retrieved successfully

## Related Information

[Location \[page 134\]](#)

### 3.1.4 Read all Matching Locations

Retrieves a list of `Location` objects matching the filter criteria provided

With this method, you send a request to the server to retrieve a subset of all locations, according to the filter criteria provided. This method offers an advanced filter functionality that goes beyond the standard filters used for collective requests of most of the other services available in the SAP Leonardo IoT platform. Here, you can define a filter criterion that consists of a comma-separated list of individual location IDs. The method then returns a collection of data sets for each `Location` object specified in the request. This method is especially useful to build a preselection of a set of locations you are interested in. For example, you might want to use this method to select all the locations in a given region or country and then perform operations on the persons or things that are assigned to these locations.



## Request

URI: `/Locations`

HTTP Method: `GET`

Permissions: `<loc>.r`

### Query String Parameters

Parameter	Required	Data Type	Description
<code>listFilterID</code>	Yes	string	comma-separated list of IDs referring to the <code>locationID</code> attribute of a <code>Location</code>

### Request Example

```
/Locations?
listFilterID=fa38e508c45743cc88dbedee2dd82464,fa38e508c45743cc88dbedee2dd82465,fa
38e508c45743cc88dbedee2dd82466
```

Retrieves three `Location` objects with the specified IDs from the database.

## Response

### Response Status and Error Codes

Code	Description
200	List of locations retrieved successfully
400	Bad request (for example, invalid filter)
403	Forbidden (user not authorized to access locations)

## Payload

Format: `JSON`

Only media type JSON is supported. The JSON for this method is an array of multiple occurrences of the following structure (one for each retrieved location):

```
{ "basicData": {
  "locationID"      : { "type": "string", "maxLength": 32 },
  "tenant"          : { "type": "string", "maxLength": 36,
"required": true },
  "etag"            : { "type": "string" }
}
"locationData": {
  "streetName"      : { "type": "string", "maxLength": 255 },
  "houseNumber"     : { "type": "string", "maxLength": 10 },
  "cityName"        : { "type": "string", "maxLength": 255 },
```

```

    "district"      : { "type": "string", "maxLength": 255 },
    "postalCode"    : { "type": "string", "maxLength": 10 },
    "country"       : { "type": "string", "maxLength": 2 },
    "countryDescription" : { "type": "string", "maxLength": 60,
"readOnly" : true },
    "region"        : { "type": "string", "maxLength": 3 },
    "regionDescription" : { "type": "string", "maxLength": 60,
"readOnly" : true },
    "longitude"     : { "type": "number"},
    "latitude"      : { "type": "number"}
  }
}

```

## Related Information

[Location \[page 134\]](#)

### 3.1.5 Update a Location

With this method, you modify a location with the specified ID.

In the payload of the request, you assign values to those fields that you want to modify. For example, you may want to correct the address data of a location that has been entered in a wrong or incomplete way. For that, you may assign values to the various fields like `streetName`, `houseNumber`, `cityName`, and so on.

## Request

**URI:** `/Locations('<locationID>')`

**HTTP Method:** *PUT*

**Permissions:** `<loc>.u`

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross site request forgery token
If-Match	Yes	Check if cached version matches server version of the Location object

### Query String Parameters

Name	Required	Data Type	Description	Parameter Type
locationID	Yes	string	Unique identifier of the Location object	Path

## Request Example

```
/Locations('4BB5CE97334A44DC850CC35062AE2B26')
```

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{ "basicData": {
  "locationID"      : { "type": "string", "maxLength": 32 },
  "tenant"          : { "type": "string", "maxLength": 36,
"required": true },
  "etag"            : { "type": "string" }
}
  "locationData": {
    "streetName"      : { "type": "string", "maxLength": 255 },
    "houseNumber"     : { "type": "string", "maxLength": 10 },
    "cityName"         : { "type": "string", "maxLength": 255 },
    "district"         : { "type": "string", "maxLength": 255 },
    "postalCode"       : { "type": "string", "maxLength": 10 },
    "country"          : { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60,
"readOnly": true },
    "region"           : { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60,
"readOnly": true },
    "longitude"        : { "type": "number"},
    "latitude"         : { "type": "number"}
  }
}
```

## Response

### Response Headers

Name	Description
Etag	Identifies the updated Location instance

### Response Status and Error Codes

Code	Description
200	Location updated successfully

## Related Information

[Location \[page 134\]](#)

## 3.1.6 Delete a Location

With this method, you delete a Location object with the specified ID.

### Request

**URI:** `/Locations('<locationID>')`

**HTTP Method:** `DELETE`

**Permissions:** `<loc>.d`

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross site request forgery token
If-Match	Yes	Check if cached version matches server version of the Location object

### Query String Parameters

Name	Required	Data Type	Description	Parameter Type
locationID	Yes	string	Unique identifier of the Location object	Path

### Request Example

```
/Locations('4BB5CE97334A44DC850CC35062AE2B26')
```

### Response

#### Response Status and Error Codes

Code	Description
204	Location deleted successfully

### Related Information

[Location \[page 134\]](#)

## 3.1.7 DeletedLocation

A `DeletedLocation` object is a physical backup for a logically deleted `Location` object.

Whenever a user deletes a `Location` object from an IoT scenario, this deletion is a logical rather than a physical deletion. That is, from a user's perspective, the deleted location is no longer available in the system and cannot be accessed anymore. However, in the background, the system takes several steps to ensure that the deleted location can still be restored, if need be. Here are some use cases where access to a previously deleted location may be required:

- Restoring a location that has accidentally been deleted
- Retrieving a location for auditing purposes
- Retrieving a location in the context of a lawsuit

Due to the special nature of the `DeletedLocation` object, access to objects of this kind is limited to system administrators with special authorizations. Moreover, a `DeletedLocation` object has a restricted set of methods that can be accessed via the Leonardo IoT API. For example, there is no `Create` method available, nor can it be modified. This is to make sure that a `DeletedLocation` objects can only be created in the system according to certain predefined and hard-coded rules. These rules enforce that `DeletedLocation` objects can only exist under well-defined circumstances laid down in legal frameworks that are relevant for operating an IoT scenario.

Write access to `DeletedLocation` objects residing in a tenant other than the one to which a user is logged on is only allowed for users who are acting in the role of a cloud platform provider.

**Base URI:** `/DeletedLocations ('<locationID>')`

**Permissions:** `<loc>.pd`

**Base URI:**

- Formal description: `https://<server address>[:<port number>]/[<path>]/DeletedLocations`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/DeletedLocations`

## Methods

HTTP Method	Action	URI	Scopes
<i>GET</i>	<a href="#">Read a DeletedLocation [page 147]</a>	<code>/DeletedLocations ('&lt;locationID&gt;')</code>	<code>&lt;loc&gt;.pd</code>
<i>GET</i>	<a href="#">Read all DeletedLocations [page 148]</a>	<code>/DeletedLocations</code>	<code>&lt;loc&gt;.pd</code>
<i>DELETE</i>	<a href="#">Delete a DeletedLocation [page 149]</a>	<code>/DeletedLocations ('&lt;locationID&gt;')</code>	<code>&lt;loc&gt;.pd</code>

## i Note

As already mentioned above, the `DeletedLocation` object has neither a POST nor a PUT method:

- There is no way of explicitly creating a `DeletedLocation` object by a method call via the interface. Instead, a `DeletedLocation` object is created automatically by the system whenever a `Location` object is deleted.
- Once the system has created a `DeletedLocation` object, there is no way of modifying its field values, neither explicitly nor implicitly. This is due to the strict legal requirements that are applicable in the use cases outlined above.

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for these methods has the following structure:

```
{ "basicData": {
    "locationID"      : { "type": "string", "maxLength": 32 },
    "tenant"          : { "type": "string", "maxLength": 36,
"required": true },
    "etag"            : { "type": "string" }
  },
  "locationData": {
    "streetName"      : { "type": "string", "maxLength": 255 },
    "houseNumber"     : { "type": "string", "maxLength": 10 },
    "cityName"         : { "type": "string", "maxLength": 255 },
    "district"         : { "type": "string", "maxLength": 255 },
    "postalCode"       : { "type": "string", "maxLength": 10 },
    "country"          : { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60,
"readOnly": true },
    "region"           : { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60,
"readOnly": true },
    "longitude"        : { "type": "number"},
    "latitude"         : { "type": "number"}
  }
}
```

## Related Information

[Location \[page 134\]](#)

## 3.1.7.1 Read a DeletedLocation

Retrieves the specified DeletedLocation object from the database

With this method, you send a request to the server to retrieve a particular DeletedLocation object specified by its ID.

If the DeletedLocation object is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

### Request

**URI:** /DeletedLocations('<locationID>')

**HTTP Method:** *GET*

**Permissions:** <loc>.pd

### Request Header Fields

Name	Required	Description
If-Match	No	Check if cached version matches server version of the location

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
locationID	Yes	string	Technical ID of the DeletedLocation object	Path

### Request Example

```
/DeletedLocations('27e663e844fe4182a387577c3e8bc6ab')
```

### Response

#### Format

**Format:** *JSON*

### Response Status and Error Codes

Code	Reason
200	DeletedLocation retrieved successfully

## Related Information

[DeletedLocation](#) [page 145]

### 3.1.7.2 Read all DeletedLocations

Retrieves a list of deleted locations from the database

With this method, you send a request to the server to retrieve a subset of all deleted locations, according to the filter criteria provided. For a collective request for a set of deleted locations, you have these options:

- You can define that the set of matching deleted locations shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

## Request

**URI:** /DeletedLocations

**HTTP Method:** [GET](#)

**Permissions:** <loc>.pd

## Query String Parameters

Name	Required	Data Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set
\$orderby	No	string	Field name to be used for sorting the result set in ascending order
\$filter	No	string	Filter condition to be applied; valid fields are country, region, postalCode, longitude, and latitude.

Allowed Combinations of Fields and Operators in Filter Conditions

Fields	Operators
longitude, latitude	eq, ne, lt, le, gt, ge
postalCode, country, region	eq, ne

## Request Example

```
/DeletedLocations
```



Retrieves all deleted locations that are available in the database.

```
/DeletedLocations?$top=25&$orderby='locationData/country'
```

Retrieves the first 25 deleted locations. Data is retrieved in page mode to reduce network load and response times, and the retrieved deleted locations are sorted by country in ascending order.

```
/DeletedLocations?$skip=10&$top=20
```

Retrieves 20 deleted locations from the database where the first 10 deleted locations are omitted. That is, the retrieved deleted locations are those with sequence numbers 11 through 30.

## Response

### Response Status and Error Codes

Code	Reason
200	Deleted locations retrieved successfully

## Related Information

[DeletedLocation](#) [page 145]

### 3.1.7.3 Delete a DeletedLocation

With this method, you physically delete a `Location` with the specified ID.

Although "deleting a deleted location" may sound strange at first glance, this is exactly the purpose of this method. The `DeletedLocation` concept represents a location that has played an active part in the system in the past, but has been logically deleted. As opposed to the logical deletion of a `Location` object, deleting a `DeletedLocation` object means:

- **explicit** physical deletion of the `DeletedLocation` object, and
- **implicit** physical deletion of the `Location` object that is represented by the corresponding `DeletedLocation` object

In other words, deleting a `DeletedLocation` object is the ultimate and irreversible way of removing all information about a particular location from the system.

#### i Note

Be very careful before executing this method. Since it is the very purpose of the `DeletedLocation` concept for you to be able to reconstruct the information stored for a location even after that location has been deleted from the system, we highly recommend that you make yourself aligned with all the relevant departments in your company before you delete a `DeletedLocation` object.

## Request

**URI:** /DeletedLocations('<locationID>')

**HTTP Method:** *DELETE*

**Permissions:** <loc>.pd

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token
If-Match	Yes	Check if cached version matches server version of the DeletedLocation object

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPartnerID	Yes	string	technical ID of the DeletedLocation object	Path

### Request Example

```
/DeletedLocations('27e663e844fe4182a387577c3e8bc6ab')
```

## Response

### Response Status and Error Codes

Code	Reason
204	DeletedLocation deleted successfully

## Related Information

[DeletedLocation \[page 145\]](#)

[Delete a Location \[page 144\]](#)

## 3.2 Geolocations Overview

A Geolocation is a geographical location of a device or an object.

The Geolocation services enable you to locate a specific object or device based on the geographical coordinates. A Geolocation has the following attributes:

- Geo-coordinates representing a point or an area
- Space representing Geolocation metadata

Geolocation services provide two types of Geolocations:

- Point of Interest (POI)
- Area of Interest (AOI)

### *Point of Interest (POI)*

A POI represents a geographical location on a map that can be associated to a Thing or to other business data, for example, Business Partner. A Point of Interest is a geographical location represented in terms of longitude and latitude. A Point of Interest can be a dealer location, store, building, etc.

You can use this data in a map by adding it as a point of interest layer, or by creating a map filter to shortlist the locations within the vicinity of another point of interest in the map.

### *Area of Interest (AOI)*

An AOI represents a geographical region on a map that can be used to analyze the Things or other business data, for example, Business Partner. You can use the geolocation services to analyze with reference to business data. An Area of Interest contains an array of coordinates, which is a polygon (three or more points). An Area of Interest can be a factory site, service center, highway, shipyard, etc.

### 3.2.1 Managing Geolocation Semantics

It provides you with information about the various operations that you can perform using geolocation semantics such as Spaces, Geometry types, GeoTest, Space Hierarchies, Location Hierarchies, and Locales.

A Geolocation can be represented in the form of a point or polygon using the geo-coordinates represented by longitude and latitude. You must add a space to a geolocation in order to define the GeoLocation geometry and enable the categorization and tagging of the related location accurately. Therefore, you must perform the following:

- Add a space to the geolocation.
- Define the geometry of the geolocation.
- Enable the categorization and tagging of the related location.

For example, Asia, India, Bangalore, Coffee Day Whitefield, Café 5.

Example Geolocation and Space

Geolocation	Space
Asia	Continent

Geolocation	Space
India	Country
Bangalore	City
Coffee Day Whitefield	Café
Café 5	Canteen

## Related Information

[Geolocation Spaces \[page 155\]](#)

[Geometry Types \[page 153\]](#)

[Locales \[page 152\]](#)

[Managing Geolocations \[page 164\]](#)

## 3.2.2 Locales

It provides you with information about the various operations that you can perform using the locale descriptions.

### *Internationalization*

Geolocation services support only geolocation descriptions or texts. If no language is selected, then by default the the locales are retrieved in English language.

### *Localization*

Geolocation services support only geolocation descriptions or texts. If no language is selected, then by default the locales are retrieved in English language.

### 3.2.2.1 Retrieve the list of Localized descriptions

This method allows you to retrieve the list of locale descriptions or texts in the geolocations service. It supports only geolocation descriptions. If no language is selected, then by default the locales are retrieved in English language.

**Base URI:** `https://<server address>/geolocation/v1/Locales`

## Request

**URI:** /geolocation/v1/Locales

**HTTP:** [GET](#)

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

## Request Parameters

None

## Request Example

```
/geolocation/v1/Locales
```

## Response

Format: [JSON](#)

## Sample Response:

≡ Output Code

```
[
  {
    "Code": "string",
    "Language": "string"
  }
]
```

Response Status and Error Codes

Code	Description
200	Retrieved the list of locales successfully

## 3.2.3 Geometry Types

It provides you with the information on the various operations that you can perform using geometry types.

The Geolocation services support two types of geometries:

- Point of Interest (POI) - The geographical point that you select on the map in Longitude and Latitude.
- Area of Interest (AOI) - The geographical region that you select on the map should be a Polygon, which is three or more points (an array of coordinates).

## Related Information

[Retrieve the list of Geometry types \[page 154\]](#)

### 3.2.3.1 Retrieve the list of Geometry types

This method allows you to retrieve the list of geometry types in the geolocation service.

**Base URI:** `https://<server address>/geolocation/v1/Geometries`

#### Request

**URI:** `/geolocation/v1/Geometries`

**HTTP:** *GET*

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

#### Request Parameters

None

#### Request Example

```
/geolocation/v1/Geometries
```

#### Response

Format: **JSON**

**Sample Response:**

 Output Code

```
[
  {
    "Type": "string"
  }
]
```

Code	Description
200	Retrieved the list of geometry types successfully

## 3.2.4 Geolocation Spaces

A Geolocation Space represents the semantics such as geometry, related geolocations, and hierarchy of a geolocation. A geolocation space should be assigned to a geolocation. For example, a Manufacturing Plant as a geolocation space can be associated to a location Xyz Manufacturing Plant, Mumbai, India. A geolocation space is represented by a unique label and a geometry type. The geolocation services allow you to list the geolocations associated with a geolocation space. You can also see the various operations that you can perform using geolocation spaces.

**Resource Path:** `http://<server address>/geolocation/v1/Spaces`

### Operations

#### CRUD Operations

HTTP Method	Operation	URI	Scopes
<i>POST</i>	<a href="#">Create a Space [page 155]</a>	<code>/geolocation/v1/Spaces</code>	<code>&lt;geoloca- tion&gt;.c</code>
<i>GET</i>	<a href="#">Retrieve the list of Spaces [page 158]</a>	<code>/geolocation/v1/Spaces</code>	<code>&lt;geolocation&gt;.r</code>
<i>GET</i>	<a href="#">Retrieve a Space [page 160]</a>	<code>/geolocation/v1/Spaces/{SpaceId}</code>	<code>&lt;geolocation&gt;.r</code>
<i>PUT</i>	<a href="#">Update a Space [page 161]</a>	<code>/geolocation/v1/Spaces/{SpaceId}</code>	<code>&lt;geoloca- tion&gt;.u</code>
<i>DELETE</i>	<a href="#">Delete a Space [page 163]</a>	<code>/geolocation/v1/Spaces/{SpaceId}</code>	<code>&lt;geoloca- tion&gt;.d</code>

### 3.2.4.1 Create a Space

This method allows you to create a geolocation space. A space is associated to a geometry, which can be a point or polygon. The geolocations of a specific geometry can be associated only to those spaces of the respective geometries.

**Base URI:** `https://<server address>/geolocation/v1/Spaces`

## Request

**URI:** /geolocation/v1/Spaces

**HTTP:** *POST*

**Roles Required:** Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

### Request Parameters

None.

### Request Example

```
/geolocation/v1/Spaces
```

## Payload

Format: **JSON**

### Note

JSON is the only media type that is supported.

## Request Properties

Property	Data Type	Required	Maximum Length	Description
Space Name	string	Required	100	The unique name of the geolocation space.
Descriptions	array	Optional	-	The description of the geolocation space.
Geometry Type	string	Optional	30	The geometry type of the geolocation space. For example, Point or Polygon.
ParentSpaceId	string	Yes	36	The parent space ID.

## Descriptions

Property	Data Type	Required	Maximum Length	Description
Label	string	Yes	200	The label name of the geolocation space.
Locale	string	Yes	2	The locale of the geolocation space. For example, EN.



## Geometry

Property	Data Type	Required	Maximum Length	Description
Type	string	Yes	20	The geometry type of the geolocation space.  For example, Point or Polygon.

The JSON for this method has the following structure:

### Code Syntax

```
[
  {
    "Descriptions": [
      {
        "Label": "string",
        "Locale": "string"
      }
    ],
    "Geometry": {
      "Type": "string"
    },
    "ParentSpaceId": "string",
    "SpaceName": "string"
  }
]
```

## Response

Response Status and Error Codes

Code	Description
201	New resource is returned in Location header
400	Invalid request. Wrong format or structure of the provided request body
403	Access denied. You did not have the required permissions to access the resource.
409	Conflict. Geolocation space already exists.
500	Internal server error. The operation you requested led to an error during execution.

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

## 3.2.4.2 Retrieve the list of Spaces

This method allows you to retrieve the list of geolocation spaces. You can filter or categorize the spaces by using geometry, name, or parent ID. This method also lets you filter geolocations by using multiple IDs. The geolocation spaces are pagination enabled by default.

**Base URI:** `https://<server address>/geolocation/v1/Spaces`

### Request

**URI:** `/geolocation/v1/Spaces`

**HTTP:** [GET](#)

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

### Request Parameters

Name	Required	Data Type	Description
\$skip	Optional	integer	The number of entities to be skipped. The default value is 0.
\$top	Optional	integer	The number of entities to be returned. The default value is 10.
SpaceName	Optional	string	The name of the geolocation space to be returned.
Geometry	Optional	string	The geometry name of the geolocation space to be returned.
ParentSpaceId	Optional	string	The Parent Space ID to be returned.
<b>Note</b> To filter root spaces, i.e., the spaces which do not have a parent, you have to search with the ParentSpaceId value as <i>None</i> .			
SpaceId	Optional	array[string]	The Space ID to be returned. It supports multiple IDs.

## Request Header

Name	Required	Data Type	Description
Accept-Language	Yes	string	Accepted language header

## Request Example

```
/geolocation/v1/Spaces
```

## Response

Format: JSON

### Sample Response:

#### Output Code

```
[
  {
    "Descriptions": [
      {
        "Label": "string",
        "Locale": "string"
      }
    ],
    "Geometry": {
      "Type": "string"
    },
    "ParentSpaceId": "string",
    "SpaceId": "string",
    "SpaceName": "string"
  }
]
```

## Response Headers

The geolocation spaces are pagination enabled by default. If you have to go to the next page or previous page, you have to use the following Link Headers: `</products/pages/1>; rel="previous"`, `</products/pages/3>; rel="next"`.

### ❖ Example

- Link: `<v1/Spaces?page=6>; rel="next"`, `<v1/Spaces?page=4>; rel="previous"`
- Link: `<v1/Spaces?page=6&size=10>; rel="next"`, `<v1/Spaces?page=4&size=10>; rel="previous"`

## Response Status and Error Codes

Code	Description
200	Retrieved the list of Geolocation spaces successfully.

Code	Description
400	Invalid request. Wrong format or structure of the provided request.
403	Access denied. You did not have the required permissions to access the resource.
404	Geolocation Space IDs not found.

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

### 3.2.4.3 Retrieve a Space

This method allows you to retrieve a particular geolocation space.

**Base URI:** `https://<server address>/geolocation/v1/Spaces/{SpaceId}`

#### Request

**URI:** `/geolocation/v1/Spaces/{SpaceId}`

**HTTP:** [GET](#)

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

Request Parameters

Name	Required	Data Type	Description
SpaceId	Yes	string	ID of the Geolocation space to be returned.

#### Request Header

Name	Required	Data Type	Description
Accept-Language	Yes	string	Accepted language header

#### Request Example

```
/geolocation/v1/Spaces/{SpaceId}
```

## Response

Format: **JSON**

**Sample Response:**

≡, Output Code

```
{
  "Descriptions": [
    {
      "Label": "string",
      "Locale": "string"
    }
  ],
  "Geometry": {
    "Type": "string"
  },
  "ParentSpaceId": "string",
  "SpaceId": "string",
  "SpaceName": "string"
}
```

Response Status and Error Codes

Code	Description
200	Retrieved the Geolocation space successfully
400	Invalid request. Wrong format or structure of the provided request.
403	Access denied. You did not have the required permissions to access the resource.
404	Geolocation Space ID not found.
500	Internal server error. The operation you requested led to an error during execution.

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

### 3.2.4.4 Update a Space

This method allows you to update an existing geolocation space with a new geolocation space.

**Base URI:** `https://<server address>/geolocation/v1/Spaces/{SpaceId}`

## Request

**URI:** `/geolocation/v1/Spaces/{SpaceId}`

HTTP: *PUT*

**Roles Required:** Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

### Request Parameters

Name	Required	Data Type	Description
SpaceId	Yes	string	ID of the Geolocation space to be updated.

### Request Example

```
/geolocation/v1/Spaces/{SpaceId}
```

## Payload

Format: **JSON**

#### Note

JSON is the only media type that is supported.

The JSON for this method has the following structure:

#### Code Syntax

```
[
  {
    "Descriptions": [
      {
        "Label": "string",
        "Locale": "string"
      }
    ],
    "Geometry": {
      "Type": "string"
    },
    "ParentSpaceId": "string",
    "SpaceName": "string"
  }
]
```

## Response

Format: **JSON**

## Response Status and Error Codes

Code	Description
204	Updated the Geolocation Space successfully
400	Invalid request. Wrong format or structure of the provided request body
403	Access denied. You did not have the required permissions to access the resource.
404	Geolocation Space ID not found.
409	Conflict. Geolocation space already exists.
500	Internal server error. The operation you requested led to an error during execution.

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

### 3.2.4.5 Delete a Space

This method allows you to delete an existing geolocation space.

**Base URI:** `https://<server address>/geolocation/v1/Spaces/{SpaceId}`

## Request

**URI:** `/geolocation/v1/Spaces/{SpaceId}`

**HTTP:** *DELETE*

**Roles Required:** Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

## Request Parameters

Name	Required	Data Type	Description
SpaceId	Yes	string	ID of the Geolocation space to be deleted.

## Request Example

```
/geolocation/v1/Spaces/{SpaceId}
```

## Response

Response Status and Error Codes

Code	Description
204	Deleted the Geolocation Space successfully
400	Invalid request. Wrong format or structure of the provided request.
403	Access denied. You did not have the required permissions to access the resource.
404	Geolocation Space ID not found.

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

## 3.2.5 Managing Geolocations

Managing Geolocations provides you with information about the various operations that you can perform using Geolocations. It allows you to create, retrieve, update, and delete Geolocations based on your requirements.

**Resource Path:** `http://<server address>/geolocation/v1/GeoLocations`

### Operations

#### CRUD Operations

HTTP Method	Operation	URI	Scopes
<i>POST</i>	<a href="#">Create a Geolocation [page 165]</a>	<code>/geolocation/v1/GeoLocations</code>	<code>&lt;geoloca- tion&gt;.c</code>
<i>GET</i>	<a href="#">Retrieve the list of Geolocations [page 168]</a>	<code>/geolocation/v1/GeoLocations</code>	<code>&lt;geolocation&gt;.r</code>
<i>GET</i>	<a href="#">Retrieve a Geolocation [page 171]</a>	<code>/geolocation/v1/GeoLocations/ {GeoLocationId}</code>	<code>&lt;geolocation&gt;.r</code>
<i>PUT</i>	<a href="#">Update a Geolocation [page 174]</a>	<code>/geolocation/v1/GeoLocations/ {GeoLocationId}</code>	<code>&lt;geoloca- tion&gt;.u</code>
<i>PUT</i>	<a href="#">Update a Geolocation (Patch) [page 176]</a>	<code>/geolocation/v1/GeoLocations</code>	<code>&lt;geoloca- tion&gt;.u</code>
<i>DELETE</i>	<a href="#">Delete a Geolocation [page 180]</a>	<code>/geolocation/v1/GeoLocations/ {GeoLocationId}</code>	<code>&lt;geoloca- tion&gt;.d</code>



### 3.2.5.1 Create a Geolocation

This method allows you to create a geolocation entry with details such as Name, Address, Description, Geometry, and Space. You can create single or multiple geolocations in the same call, and this method lets you create upto 20 geolocations in a single call.

Geolocation services support two Geometry Types:

- Point
- Polygon

For more information, see [Geometry Types \[page 153\]](#)

**Base URI:** `https://<server address>/geolocation/v1/GeoLocations`

#### Request

**URI:** `/geolocation/v1/GeoLocations`

**HTTP:** *POST*

**Roles Required:** Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

#### Request Parameters

None.

#### Request Example

```
/geolocation/v1/GeoLocations
```

#### Payload

Format: **JSON**

##### Note

JSON is the only media type that is supported.

#### Request Properties

##### Properties

Property	Data Type	Required	Maximum Length	Description
SpaceId	string	Yes	36	The space ID of the geolocation.

Property	Data Type	Required	Maximum Length	Description
ExternalId	string	Optional	60	The external ID of the geolocation.
GeoLocation-Name	string	Yes	100	The unique name of the geolocation.
ParentGeolocationId	string	Optional	36	The parent geolocation ID.
Descriptions	array	Optional	-	The description of the geolocation.
Addresses	array	Optional	-	The address of the geolocation.

## Descriptions

Property	Data Type	Required	Maximum Length	Description
Label	string	Yes	200	The description label of the geolocation.
Locale	string	Yes	2	The locale of the geolocation. For example, EN.

## Addresses

Property	Data Type	Required	Maximum Length	Description
Address	array	Yes	-	The address of the geolocation.
Locale	string	Yes	2	The locale of the geolocation. For example, EN.

## Address

Property	Data Type	Required	Maximum Length	Description
House	string	Optional	10	The house number of the geolocation.
Street	string	Optional	40	The street details of the geolocation.
CityDistrict	string	Optional	40	The city district of the geolocation.
City	string	Optional	40	The city of the geolocation.
District	string	Optional	40	The district of the geolocation.
RegionCode	string	Optional	3	The region code of the geolocation.

Property	Data Type	Required	Maximum Length	Description
Region	string	Optional	40	The region of the geolocation.
CountryCode	string	Optional	3	The country code of the geolocation.
Country	string	Optional	40	The country of the geolocation.
PostalCode	string	Optional	10	The postal code of the geolocation.

## Geometry

Property	Data Type	Required	Maximum Length	Description
type	string	Yes	20	The geometry type of the geolocation.  For example, Point or Polygon.
coordinates	GeoJSON	Yes	-	The coordinates of the geolocation, which is longitude and latitude. For example, {"type": "Point", "coordinates": [longitude, latitude]}.

The JSON for this method has the following structure:

### Code Syntax

```
[
  {
    "Geometry": {
      "type": "string",
      "coordinates": [ ]
    },
    "Properties": {
      "Addresses": [
        {
          "Address": {
            "City": "string",
            "CityDistrict": "string",
            "Country": "string",
            "CountryCode": "string",
            "District": "string",
            "House": "string",
            "PostalCode": "string",
            "Region": "string",
            "RegionCode": "string",
            "Street": "string"
          },
          "Locale": "string"
        }
      ],
      "Descriptions": [
        {
          "Label": "string",
          "Locale": "string"
        }
      ]
    }
  }
]
```

```

    }
  ],
  "ExternalId": "string",
  "GeoLocationName": "string",
  "ParentGeoLocationId": "string",
  "SpaceId": "string"
},
"Type": "string"
}
]

```

## Response

Format: **JSON**

Response Status and Error Codes

Code	Description
201	New resource is returned in Location header
400	Invalid request. Wrong format or structure of the provided request body.
403	Access denied. You did not have the required permissions to access the resource.
500	Internal server error. The operation you requested led to an error during execution.

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

### 3.2.5.2 Retrieve the list of Geolocations

This method allows you to retrieve the list of existing geolocations. You can filter or categorize the geolocations by using Space, Geometry, Name, External ID, or Parent ID. This method also lets you filter geolocations by using multiple IDs.

**Base URI:** `https://<server address>/geolocation/v1/GeoLocations`

## Request

**URI:** `/geolocation/v1/GeoLocations`

**HTTP:** *GET*

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

## Request Parameters

Name	Required	Data Type	Description
\$skip	Optional	integer	The number of entities to be skipped. The default value is 0.
\$top	Optional	integer	The number of entities to be returned. The default value is 10.
SpaceName	Optional	string	The name of the space to be returned.
Geometry	Optional	string	The name of the geometry to be returned.
GeoLocationName	Optional	string	The name of the geolocation to be returned.
ExternalId	Optional	string	The External ID of the geolocation to be returned.
ParentGeoLocationId	Optional	string	The Parent geolocation ID to be returned.
<div><b>i Note</b> To filter root geolocations, which are geolocations that do not have a parent, you have to search with the ParentGeoLocationId value as <i>none</i>.</div>			
GeoLocationId	Optional	array[string]	The geolocation ID to be returned. It supports multiple IDs.
SpaceId	Optional	array[string]	The space ID to be returned. It supports multiple IDs.

## Request Header

Name	Required	Data Type	Description
Accept-Language	Yes	string	Accepted language header

## Request Example

```
/geolocation/v1/GeoLocations
```

## Response

Format: **JSON**

**Sample Response:**

≡ Output Code

```
[
  {
    "Geometry": {
      "type": "string",
      "coordinates": [ ]
    },
    "Properties": {
      "Addresses": [
        {
          "Address": {
            "City": "string",
            "CityDistrict": "string",
            "Country": "string",
            "CountryCode": "string",
            "District": "string",
            "House": "string",
            "PostalCode": "string",
            "Region": "string",
            "RegionCode": "string",
            "Street": "string"
          },
          "Locale": "string"
        }
      ],
      "Descriptions": [
        {
          "Label": "string",
          "Locale": "string"
        }
      ],
      "ExternalId": "string",
      "GeoLocationId": "string",
      "GeoLocationName": "string",
      "ParentGeoLocationId": "string",
      "SpaceId": "string",
      "SpaceName": "string"
    },
    "Type": "string"
  }
]
```

## Response Headers

The geolocation spaces are pagination enabled by default. If you have to go to the next page or previous page, you must use the following Link Headers: `</products/pages/1>; rel="previous"`, `</products/pages/3>; rel="next"`.

### ❁ Example

- Link: `<v1/GeoLocations?page=6>; rel="next"`, `<v1/GeoLocations?page=4>; rel="previous"`
- Link: `<v1/GeoLocations?page=6&size=10>; rel="next"`, `<v1/GeoLocations?page=4&size=10>; rel="previous"`

#### Response Status and Error Codes

Code	Description
200	Retrieved the list of Geolocations successfully
400	Invalid request. Wrong format or structure of the provided request.
403	Access denied. You did not have the required permissions to access the resource.

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

### 3.2.5.3 Retrieve a Geolocation

This method allows you to retrieve the list of existing geolocations. You can filter or categorize the geolocations by using Space, Geometry, Name, External ID, or Parent ID. This method also lets you filter geolocations by using multiple IDs.

**Base URI:** `https://<server address>/geolocation/v1/GeoLocations`

#### Request

**URI:** `/geolocation/v1/GeoLocations`

**HTTP:** [GET](#)

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

#### Request Parameters

Name	Required	Data Type	Description
\$skip	Optional	integer	The number of entities to be skipped. The default value is 0.
\$top	Optional	integer	The number of entities to be returned. The default value is 10.
SpaceName	Optional	string	The name of the space to be returned.

Name	Required	Data Type	Description
Geometry	Optional	string	The name of the geometry to be returned.
GeoLocationName	Optional	string	The name of the geolocation to be returned.
ExternalId	Optional	string	The External ID of the geolocation to be returned.
ParentGeoLocationId	Optional	string	The Parent geolocation ID to be returned.
			<b>i Note</b> To filter root geolocations, which are geolocations that do not have a parent, you have to search with the ParentGeoLocationId value as <i>none</i> .
GeoLocationId	Optional	array[string]	The geolocation ID to be returned. It supports multiple IDs.
SpaceId	Optional	array[string]	The space ID to be returned. It supports multiple IDs.

## Request Header

Name	Required	Data Type	Description
Accept-Language	Yes	string	Accepted language header

## Request Example

```
/geolocation/v1/GeoLocations
```

## Response

Format: **JSON**

**Sample Response:**

≡ Output Code

```
[
  {
    "Geometry": {
```



```

    "type": "string",
    "coordinates": [ ]
  },
  "Properties": {
    "Addresses": [
      {
        "Address": {
          "City": "string",
          "CityDistrict": "string",
          "Country": "string",
          "CountryCode": "string",
          "District": "string",
          "House": "string",
          "PostalCode": "string",
          "Region": "string",
          "RegionCode": "string",
          "Street": "string"
        },
        "Locale": "string"
      }
    ],
    "Descriptions": [
      {
        "Label": "string",
        "Locale": "string"
      }
    ],
    "ExternalId": "string",
    "GeoLocationId": "string",
    "GeoLocationName": "string",
    "ParentGeoLocationId": "string",
    "SpaceId": "string",
    "SpaceName": "string"
  },
  "Type": "string"
}
]

```

## Response Headers

The geolocation spaces are pagination enabled by default. If you have to go to the next page or previous page, you must use the following Link Headers: `</products/pages/1>; rel="previous"`, `</products/pages/3>; rel="next"`.

### ❖ Example

- Link: `<v1/GeoLocations?page=6>; rel="next"`, `<v1/GeoLocations?page=4>; rel="previous"`
- Link: `<v1/GeoLocations?page=6&size=10>; rel="next"`, `<v1/GeoLocations?page=4&size=10>; rel="previous"`

## Response Status and Error Codes

Code	Description
200	Retrieved the list of Geolocations successfully
400	Invalid request. Wrong format or structure of the provided request.

Code	Description
403	Access denied. You did not have the required permissions to access the resource.

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

## 3.2.5.4 Update a Geolocation

This method allows you to update an existing geolocation with a new geolocation, but you are not allowed to change the Geometry Type.

**Base URI:** `https://<server address>/geolocation/v1/GeoLocations/{GeoLocationId}`

### Request

**URI:** `/geolocation/v1/GeoLocations/{GeoLocationId}`

**HTTP:** *PUT*

**Roles Required:** Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

### Request Parameters

Name	Required	Data Type	Description
GeoLocationId	Yes	string	ID of the Geolocation to be updated

### Request Example

```
/geolocation/v1/GeoLocations/{GeoLocationId}
```

### Payload

Format: **JSON**

#### i Note

JSON is the only media type that is supported.

The JSON for this method has the following structure:

#### Code Syntax

```
[
  {
    "Geometry": {
      "type": "string",
      "coordinates": [ ]
    },
    "Properties": {
      "Addresses": [
        {
          "Address": {
            "City": "string",
            "CityDistrict": "string",
            "Country": "string",
            "CountryCode": "string",
            "District": "string",
            "House": "string",
            "PostalCode": "string",
            "Region": "string",
            "RegionCode": "string",
            "Street": "string"
          },
          "Locale": "string"
        }
      ],
      "Descriptions": [
        {
          "Label": "string",
          "Locale": "string"
        }
      ],
      "ExternalId": "string",
      "GeoLocationName": "string",
      "ParentGeoLocationId": "string",
      "SpaceId": "string",
      "SpaceName": "string"
    },
    "Type": "string"
  }
]
```

## Response

Format: **JSON**

Response Status and Error Codes

Code	Description
204	Updated the Geolocation successfully
400	Invalid request. Wrong format or structure of the provided request body.

Code	Description
403	RequestAccess denied. You did not have the required permissions to access the resource.
404	Geolocation ID not found.
500	Internal server error. The operation you requested led to an error during execution.

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

### 3.2.5.5 Update a Geolocation (Patch)

This method allows you to update a geolocation entry with details such as Name, Address, Description, Geometry, and Space. You can update single or multiple geolocations in the same call, and this method lets you update upto 20 geolocations in a single call.

Geolocation services support two Geometry Types:

- Point
- Polygon

For more information, see [Geometry Types \[page 153\]](#)

**Base URI:** `https://<server address>/geolocation/v1/GeoLocations`

#### Request

**URI:** `/geolocation/v1/GeoLocations`

**HTTP:** *POST*

**Roles Required:** Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

#### Request Parameters

None.

#### Request Example

```
/geolocation/v1/GeoLocations
```

## Payload

Format: **JSON**

### **i** Note

JSON is the only media type that is supported.

## Request Properties

### Properties

Property	Data Type	Required	Maximum Length	Description
GeoLocationId	string	Yes	36	The ID of the geolocation.
SpaceId	string	Optional	36	The space ID of the geolocation.
ExternalId	string	Optional	60	The external ID of the geolocation.  <b>i</b> Note The value can be null.
GeoLocation-Name	string	Optional	100	The unique name of the geolocation.
ParentGeolocationId	string	Optional	36	The parent geolocation ID.  <b>i</b> Note The value can be null.
Descriptions	array	Optional	-	The description of the geolocation.  <b>i</b> Note The value can be null.
Addresses	array	Optional	-	The address of the geolocation.  <b>i</b> Note The value can be null.

## Descriptions

Property	Data Type	Required	Maximum Length	Description
Label	string	Yes	200	The description label of the geolocation.
Locale	string	Yes	2	The locale of the geolocation. For example, EN.

## Addresses

Property	Data Type	Required	Maximum Length	Description
Address	array	Yes	-	The address of the geolocation.
Locale	string	Yes	2	The locale of the geolocation. For example, EN.

## Address

### Note

The values of the following properties can be set to null. But, all the field values cannot be null at the same time. If all the values are set to null then the update request would not be successful.

Property	Data Type	Required	Maximum Length	Description
House	string	Optional	10	The house number of the geolocation.
Street	string	Optional	40	The street details of the geolocation.
CityDistrict	string	Optional	40	The city district of the geolocation.
City	string	Optional	40	The city of the geolocation.
District	string	Optional	40	The district of the geolocation.
RegionCode	string	Optional	3	The region code of the geolocation.
Region	string	Optional	40	The region of the geolocation.
CountryCode	string	Optional	3	The country code of the geolocation.
Country	string	Optional	40	The country of the geolocation.
PostalCode	string	Optional	10	The postal code of the geolocation.

## Geometry

Property	Data Type	Required	Maximum Length	Description
type	string	Yes	20	The geometry type of the geolocation.  For example, Point or Polygon.

Property	Data Type	Required	Maximum Length	Description
coordinates	GeoJSON	Yes	-	The coordinates of the geolocation, which is longitude and latitude. For example, {"type": "Point", "coordinates": [longitude, latitude]}.

The JSON for this method has the following structure:

#### Code Syntax

```
[
  {
    "Geometry": {
      "type": "string",
      "coordinates": [ ]
    },
    "Properties": {
      "Addresses": [
        {
          "Address": {
            "City": "string",
            "CityDistrict": "string",
            "Country": "string",
            "CountryCode": "string",
            "District": "string",
            "House": "string",
            "PostalCode": "string",
            "Region": "string",
            "RegionCode": "string",
            "Street": "string"
          },
          "Locale": "string"
        }
      ],
      "Descriptions": [
        {
          "Label": "string",
          "Locale": "string"
        }
      ],
      "GeoLocationId": "string",
      "ExternalId": "string",
      "GeoLocationName": "string",
      "ParentGeoLocationId": "string",
      "SpaceId": "string"
    },
    "Type": "string"
  }
]
```

## Response

Format: **JSON**

## Response Status and Error Codes

Code	Description
204	Updated the Geolocation successfully
400	Invalid request. Wrong format or structure of the provided request body.
403	Access denied. You did not have the required permissions to access the resource.
500	Internal server error. The operation you requested led to an error during execution.

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

## 3.2.5.6 Delete a Geolocation

This method allows you to delete an existing geolocation. You can enter the ID of the geolocation that you want to delete, and you can request for deletion of the geolocation. If the geolocation does not have any dependencies, only then the geolocation gets deleted.

**Base URI:** `https://<server address>[:<port number>][/path]/geolocation/v1/GeoLocations/{GeoLocationId}`

## Request

**URI:** `/geolocation/v1/GeoLocations/{GeoLocationId}`

**HTTP:** *DELETE*

**Roles Required:** Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

## Request Parameters

Name	Required	Data Type	Description
GeoLocationId	Yes	string	ID of the Geolocation to be deleted

## Request Example

```
/geolocation/v1/GeoLocations/{GeoLocationId}
```



## Response

Format: **JSON**

Response Status and Error Codes

Code	Description
204	Deleted the geolocation successfully
400	Invalid request. Wrong format or structure of the provided request.
403	Access denied. You did not have the required permissions to access the resource.
404	Geolocation ID not found.

|

For more information on the error response structures, see [Error Response Structures \[page 201\]](#)

## 3.2.6 GeoTest

GeoTest is used to perform geofencing tests on geo-coordinates or geolocation geometries. You can check things like the geolocations within the vicinity of a geolocation, or if any other geolocation is within the vicinity of a geolocation or geoposition, or if a geolocation is within a specified distance from a geoposition.

**Resource Path:** `http://<server address>/geolocation/v1/GeoTest/Check/Within/GeoLocation`

### Operations

#### CRUD Operations

HTTP Method	Operation	URI	Scopes
GET	<a href="#">Match Geolocations within a Geolocation [page 182]</a>	/geolocation/v1/GeoTest/Check/Within/GeoLocation	<geolocation>.r
GET	<a href="#">Match Geolocations within a Geoposition [page 183]</a>	/geolocation/v1/Spaces	<geolocation>.r
GET	<a href="#">Search Geolocations in close vicinity to another Geolocation [page 185]</a>	/geolocation/v1/Spaces/{SpaceId}	<geolocation>.r
GET	<a href="#">Search Geolocations within a specified distance from a Geoposition [page 187]</a>	/geolocation/v1/Spaces/{SpaceId}	<geolocation>.r

## Related Information

[Role Template and Scopes \[page 200\]](#)

### 3.2.6.1 Match Geolocations within a Geolocation

This method allows you to search for other geolocations within the vicinity of a particular geolocation using the geolocation ID. If distance is not provided, then it checks for geolocations considering the distance value as 0. The maximum distance that you can search for a geolocation is 20,000 meters.

**Base URI:** `https://<server address>/geolocation/v1/GeoTest/Check/Within/GeoLocation`

## Request

**URI:** `/geolocation/v1/GeoTest/Check/Within/GeoLocation`

**HTTP:** [GET](#)

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

## Request Parameters

Name	Required	Data Type	Description
GeoLocationId1	Yes	string	ID of the geolocation to be matched
GeoLocationId2	Yes	string	ID of the geolocation that contains the previous geolocation
Distance	Optional	double	Distance in meters. For example, 10. The default value is 0.

## Request Example

```
/geolocation/v1/GeoTest/Check/Within/GeoLocation
```

## Response

Format: **JSON**

### Sample Response:

#### Output Code

```
{
  "Distance": 0,
  "GeoLocationId1": "string",
  "GeoLocationId2": "string",
  "Within": true
}
```

#### Response Status and Error Codes

Code	Description
200	Matched geolocations within the vicinity of a geolocation successfully
400	Invalid request. Wrong format or structure of the provided request body
403	Access denied. You did not have the required permissions to access the resource.
404	Geolocation ID not found.

## 3.2.6.2 Match Geolocations within a Geoposition

This method allows you to match the geolocations within the vicinity of a Geoposition.

If distance is not provided, then it checks for geolocations considering the distance value as 0. The maximum distance that you can search for a geolocation is 20,000 metres.

#### Note

If the geoposition and the geolocation have the exact same coordinates, the geoposition check service shows that the geolocation is not in the vicinity of the geoposition. This issue occurs due to the precision while storing the geolocation geometries. To overcome this limitation and get an exact match, make sure you search with 1 or greater than 1 as the default value in the distance parameter.

**Base URI:** `https://<server address>/geolocation/v1/GeoTest/Check/Within/GeoPosition`

## Request

**URI:** `/geolocation/v1/GeoTest/Check/Within/GeoPosition`

**HTTP:** *GET*

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

## Request Parameters

Name	Required	Data Type	Description
Latitude	Yes	double	Latitude in degrees, ranges from -ve 90 to +ive 90. For example, 13.1217
Longitude	Yes	double	Longitude in degrees, ranges from -ve 180 to +ive 180. For example, 77.5804
GeoLocationId	Yes	string	ID of the geolocation. For example, WARD_2, POI_WARD_2
Distance	Optional	double	Distance in meters. For example, 10. The default value is 0.

## Request Example

```
/geolocation/v1/GeoTest/Check/Within/GeoPosition
```

## Response

Format: **JSON**

### Sample Response:

#### Output Code

```
{
  "Distance": 0,
  "GeoLocationId": "string",
  "Point": [
    long,
    lat
  ],
  "Within": true
}
```

### Response Status and Error Codes

Code	Description
200	Matched the geolocations within the vicinity of a geoposition successfully

Code	Description
400	Invalid request. Wrong format or structure of the provided request body
403	Access denied. You did not have the required permissions to access the resource.
404	Geolocation ID not found

### 3.2.6.3 Search Geolocations in close vicinity to another Geolocation

This method allows you to search for other geolocations within the vicinity of a geolocation using the geolocation ID. You can get the list of geolocations in close vicinity to the required geolocation. The maximum distance that you can search for a geolocation is 20,000 meters.

**Base URI:** `https://<server address>/geolocation/v1/GeoTest/Search/Within/GeoLocation`

#### Request

**URI:** `/geolocation/v1/GeoTest/Search/Within/GeoLocation`

**HTTP:** *GET*

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

#### Request Parameters

Name	Required	Data Type	Description
GeoLocationId	Yes	string	ID of the geolocation to be searched.
Distance	Optional	double	Distance in meters. For example, 10. The default value is 0.
SpaceId	Optional	string	ID of the geolocation space to be searched. The default value is ALL.

Name	Required	Data Type	Description
\$skip	Optional	integer	The number of entities to be skipped. The default value is 0.
\$top	Optional	integer	The number of entities to be returned. The default value is 10.

## Request Header

Request Header

Name	Required	Data Type	Description
Accept-Language	Yes	string	Accept language header

## Request Example

```
/geolocation/v1/GeoTest/Search/Within/GeoLocation
```

## Response

Format: JSON

### Sample Response

#### Output Code

```
[
  {
    "GeoLocationId": "string",
    "GeoLocations": {
      "Features": [
        {
          "Geometry": {
            "type": "string",
            "coordinates": [....]
          },
          "Properties": {
            "Addresses": [
              {
                "Address": {
                  "City": "string",
                  "CityDistrict": "string",
                  "Country": "string",
                  "CountryCode": "string",
                  "District": "string",
                  "House": "string",
                  "PostalCode": "string",
                  "Region": "string",
                  "RegionCode": "string",
                  "Street": "string"
                }
              },
              "Locale": "string"
            ]
          }
        }
      ]
    }
  }
]
```

```

    }
  ],
  "Descriptions": [
    {
      "Label": "string",
      "Locale": "string"
    }
  ],
  "ExternalId": "string",
  "GeoLocationId": "string",
  "Name": "string",
  "ParentGeoLocationId": "string",
  "SpaceId": "string",
  "SpaceName": "string"
},
"Type": "string"
}
],
"Type": "string"
},
"Space": "string"
}
]

```

#### Response Status and Error Codes

Code	Description
200	Retrieved the geolocations in close vicinity to another geolocation successfully.
400	Invalid request. Wrong format or structure of the provided request body.
403	Access denied. You did not have the required permissions to access the resource.
500	Internal server error. The operation you requested led to an error during execution.

### 3.2.6.4 Search Geolocations within a specified distance from a Geoposition

This method allows you to search for geolocations within the specified distance from a geoposition. The maximum distance that you can search for a geolocation is 20,000 meters.

**Base URI:** `https://<server address>/geolocation/v1/GeoTest/Search/Within/GeoPosition`

#### Request

**URI:** `/geolocation/v1/GeoTest/Search/Within/GeoPosition`

HTTP: [GET](#)

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

## Request Parameters

Name	Required	Data Type	Description
Latitude	Yes	double	Latitude in degrees, ranges from -ve 90 to +ive 90. For example, 13.1217
Longitude	Yes	double	Longitude in degrees, ranges from -ve 180 to +ive 180. For example, 77.5804
Distance	Optional	double	Distance in meters. For example, 10. The default value is 0.
SpaceId	Optional	string	ID of the geolocation space to be searched. The default value is ALL.
\$skip	Optional	integer	The number of entities to be skipped. The default value is 0.
\$top	Optional	integer	The number of entities to be returned. The default value is 10.

## Request Header

Name	Required	Data Type	Description
Accept-Language	Yes	string	Accept language header

## Request Example

```
/geolocation/v1/GeoTest/Search/Within/GeoPosition
```

## Response

Format: **JSON**

## Sample Response



## Output Code

```
[
  {
    "GeoLocations": {
      "Features": [
        {
          "Geometry": {
            "type": "string"
            "coordinates": [....]
          },
          "Properties": {
            "Addresses": [
              {
                "Address": {
                  "City": "string",
                  "CityDistrict": "string",
                  "Country": "string",
                  "CountryCode": "string",
                  "District": "string",
                  "House": "string",
                  "PostalCode": "string",
                  "Region": "string",
                  "RegionCode": "string",
                  "Street": "string"
                },
                "Locale": "string"
              }
            ],
            "Descriptions": [
              {
                "Label": "string",
                "Locale": "string"
              }
            ],
            "ExternalId": "string",
            "GeoLocationId": "string",
            "Name": "string",
            "ParentGeoLocationId": "string",
            "SpaceId": "string",
            "SpaceName": "string"
          },
          "Type": "string"
        }
      ],
      "Type": "string"
    },
    "Point": [
      long,
      lat
    ],
    "Space": "string"
  }
]
```

## Response Status and Error Codes

Code	Description
200	Retrieved the geolocations within specified distance from a geoposition

Code	Description
400	Invalid request. Wrong format or structure of the provided request body
403	Access denied. You did not have the required permissions to access the resource.
500	Internal server error. The operation you requested led to an error during execution.

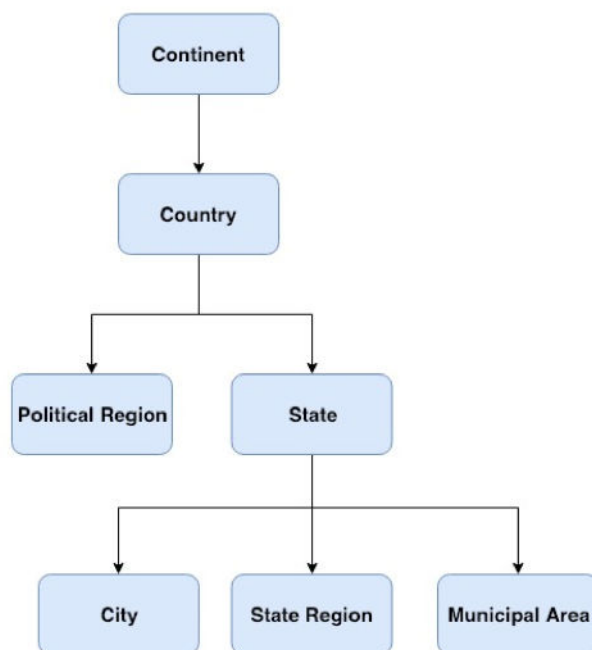
## 3.2.7 Space Hierarchies and Location Hierarchies

### Space Hierarchies

A mechanism that helps in building locations hierarchy based on a reference space hierarchy representation. Space hierarchy can be built only for spaces with geometry type as polygon. Space hierarchies have the following criteria:

- A space can be present only in one space hierarchy and it can't be a part of multiple space hierarchies.
- A space can have only one parent space.
- A space can have multiple children spaces.
- You can create any number of levels for your space hierarchies.

For example, you can see the Continent Space Hierarchy:

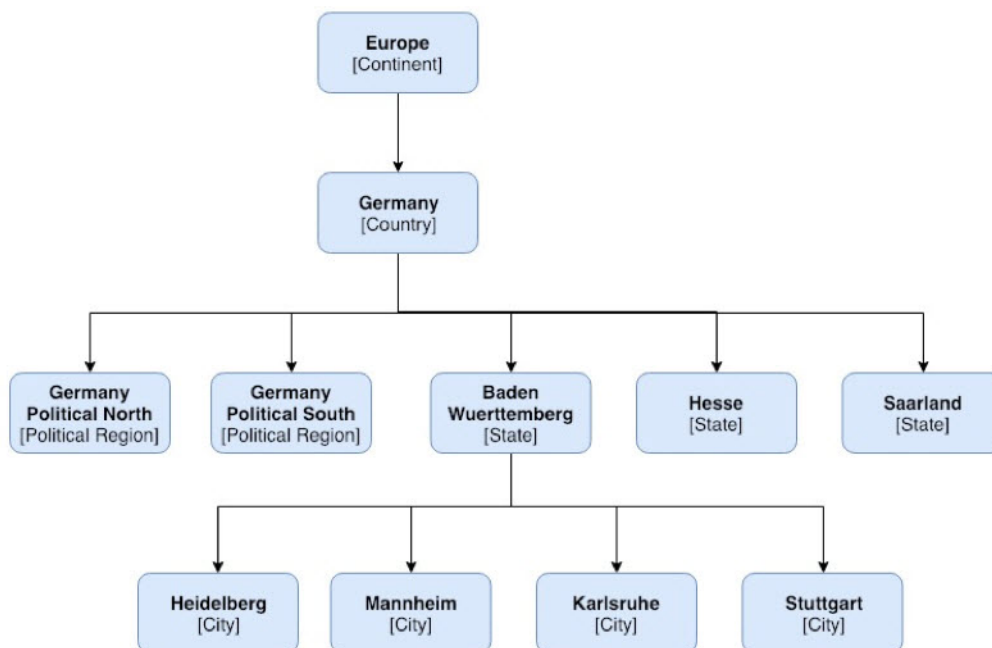


### Location Hierarchies

A mechanism that helps in organizing hierarchy based on the space hierarchy. This mechanism of organizing the locations based on a space hierarchy, enables us to maintain and follow the semantic hierarchy. Location hierarchies have the following criteria:

- A location can refer to a specific node of a space hierarchy based on the association of space with the location.
- To make a location as a part of a specific space hierarchy, you must:
  - Associate a space to a location
  - Specify the parent location of the current location
- Specifying a parent location is optional. But, if the parent location is not specified, then the location is not a part of any location hierarchy.
- To select a parent location, make sure that the parent location is associated to a space, which is the parent space of the space associated with the current location. For example, in the following diagram, the Assembly Line Drawbar is associated with the Space [\_Assembly Line\_], therefore the parent location of this location should have been associated to the Space [\_Plant\_] as per the space hierarchy that is modeled.

For example, you can see the Continent Location Hierarchy:



## Related Information

[Role Template and Scopes \[page 200\]](#)

## 3.2.7.1 Retrieve the list of Hierarchies of Spaces

This method allows you to retrieve the list of hierarchies of spaces. It also lets you filter geolocations by using multiple IDs.

**Base URI:** `https://<server address>/geolocation/v1/Hierarchies/Spaces`

### Request

**URI:** `/geolocation/v1/Hierarchies/Spaces`

**HTTP:** [GET](#)

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

### Request Parameters

Name	Required	Data Type	Description
\$skip	Optional	integer	The number of entities to be skipped. The default value is 0.
\$top	Optional	integer	The number of entities to be returned. The default value is 10.
SpaceId	Optional	array[string]	The hierarchies of the specified Space IDs to be returned. It supports multiple IDs.

### Request Header

Request Header

Name	Required	Data Type	Description
Accept-Language	Yes	string	Accepted language header

### Request Example

```
/geolocation/v1/Hierarchies/Spaces
```

### Response

Format: **JSON**

#### Sample Response:

##### Output Code

```
[
  {
    "Children": [
      {
        "Children": [
          {}
        ],
        "Descriptions": [
          {
            "Label": "string",
            "Locale": "string"
          }
        ],
        "ParentSpaceId": "string",
        "SpaceId": "string",
        "SpaceName": "string"
      }
    ],
    "Descriptions": [
      {
        "Label": "string",
        "Locale": "string"
      }
    ],
    "ParentSpaceId": "string",
    "SpaceId": "string",
    "SpaceName": "string"
  }
]
```

#### Response Status and Error Codes

Code	Description
200	Retrieved the list of Hierarchies for Spaces successfully
400	Invalid request. Wrong format or structure of the provided request.
403	Access denied. You did not have the required permissions to access the resource.

## 3.2.7.2 Retrieve a Hierarchy of Spaces

This method allows you retrieve the hierarchy of geolocation spaces using the Space ID.

**Base URI:** `https://<server address>/geolocation/v1/Hierarchies/Spaces/{SpaceId}`

## Request

**URI:** /geolocation/v1/Hierarchies/Spaces/{SpaceId}

**HTTP:** [GET](#)

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

Name	Required	Data Type	Description
SpaceId	Yes	string	ID of the root node of space

## Request Header

Name	Required	Data Type	Description
Accept-Language	Yes	string	Accepted language header

## Request Example

```
/geolocation/v1/Hierarchies/Spaces/{SpaceId}
```

## Response

Format: **JSON**

**Sample Response:**

≡ Output Code

```
{
  "Children": [
    {}
  ],
  "Descriptions": [
    {
      "Label": "string",
      "Locale": "string"
    }
  ],
  "ParentSpaceId": "string",
  "SpaceId": "string",
  "SpaceName": "string"
}
```

## Response Status and Error Codes

Code	Description
200	Retrieved the Hierarchy of the Space successfully
400	Invalid request. Wrong format or structure of the provided request.
403	Access denied. You did not have the required permissions to access the resource.
404	Geolocation Space IDs not found.
500	Internal server error. The operation you requested led to an error during execution.

### 3.2.7.3 Retrieve the list of Hierarchies for the Locations of a Space

This method allows you to retrieve the list of hierarchies for the locations of a space using Space ID.

**Base URI:** `https://<server address>/geolocation/v1/Hierarchies/Spaces/{SpaceId}/GeoLocations`

## Request

**URI:** `/geolocation/v1/Hierarchies/Spaces/{SpaceId}/GeoLocations`

**HTTP:** [GET](#)

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

## Request Parameters

Name	Required	Data Type	Description
SpaceId	Yes	string	ID of the root node of space
GeoLocationId	Optional	array[string]	The hierarchies of the specified Geolocation IDs to be returned. It supports multiple IDs.

## Request Header

Name	Required	Data Type	Description
Accept-Language	Yes	string	Accepted language header

## Request Example

```
/geolocation/v1/Hierarchies/Spaces/{SpaceId}/GeoLocations
```

## Response

Format: JSON

Sample Response:

≡ Output Code

```
[
  {
    "Children": [
      {
        "Children": [
          {}
        ],
        "Geometry": {
          "type": "string",
          "coordinates": [ ]
        },
        "Properties": {
          "Addresses": [
            {
              "Address": {
                "City": "string",
                "CityDistrict": "string",
                "Country": "string",
                "CountryCode": "string",
                "District": "string",
                "House": "string",
                "PostalCode": "string",
                "Region": "string",
                "RegionCode": "string",
                "Street": "string"
              },
              "Locale": "string"
            }
          ],
          "Descriptions": [
            {
              "Label": "string",
              "Locale": "string"
            }
          ],
          "ExternalId": "string",
          "GeoLocationId": "string",
          "GeoLocationName": "string",
          "ParentGeoLocationId": "string",
          "SpaceId": "string",
          "SpaceName": "string"
        }
      }
    ]
  }
]
```



```

    },
    "Type": "string"
  }
],
"Geometry": {
  "type": "string",
  "coordinates": [ ]
},
"Properties": {
  "Addresses": [
    {
      "Address": {
        "City": "string",
        "CityDistrict": "string",
        "Country": "string",
        "CountryCode": "string",
        "District": "string",
        "House": "string",
        "PostalCode": "string",
        "Region": "string",
        "RegionCode": "string",
        "Street": "string"
      },
      "Locale": "string"
    }
  ],
  "Descriptions": [
    {
      "Label": "string",
      "Locale": "string"
    }
  ],
  "ExternalId": "string",
  "GeoLocationId": "string",
  "GeoLocationName": "string",
  "ParentGeoLocationId": "string",
  "SpaceId": "string",
  "SpaceName": "string"
},
"Type": "string"
}
]

```

#### Response Status and Error Codes

Code	Description
200	Retrieved the Hierarchies of the Locations of a Space successfully.
400	Invalid request. Wrong format or structure of the provided request.
403	Access denied. You did not have the required permissions to access the resource.
404	Geolocation Space IDs not found.
500	Internal server error. The operation you requested led to an error during execution.

## 3.2.7.4 Retrieve a Hierarchy for the Locations of a Space

This method allows you to retrieve a hierarchy for the locations of a space using Space ID and Geolocation ID.

**Base URI:** `https://<server address>/geolocation/v1/Hierarchies/Spaces/{SpaceId}/GeoLocations/{GeoLocationId}`

### Request

**URI:** `/geolocation/v1/Hierarchies/Spaces/{SpaceId}/GeoLocations/{GeoLocationId}`

**HTTP:** *GET*

**Roles Required:** Geolocation\_Viewer, Geolocation\_Editor, Geolocation\_Super\_Editor

For more information on Roles and Permissions, see [Role Template and Scopes \[page 200\]](#)

Name	Required	Data Type	Description
SpaceId	Yes	string	ID of the root node of space
GeoLocationId	Yes	array[string]	The hierarchies of the specified Geolocation IDs to be returned. It supports multiple IDs.

### Request Header

Name	Required	Data Type	Description
Accept-Language	Yes	string	Accepted language header

### Request Example

```
/geolocation/v1/Hierarchies/Spaces/{SpaceId}/GeoLocations
```

### Response

Format: **JSON**

**Sample Response:**

≡ Output Code

```
{
  "Children": [
    {}
  ],
  "Geometry": {
```

```

    "type": "string"
  },
  "Properties": {
    "Addresses": [
      {
        "Address": {
          "City": "string",
          "CityDistrict": "string",
          "Country": "string",
          "CountryCode": "string",
          "District": "string",
          "House": "string",
          "PostalCode": "string",
          "Region": "string",
          "RegionCode": "string",
          "Street": "string"
        },
        "Locale": "string"
      }
    ],
    "Descriptions": [
      {
        "Label": "string",
        "Locale": "string"
      }
    ],
    "ExternalId": "string",
    "GeoLocationId": "string",
    "GeoLocationName": "string",
    "ParentGeoLocationId": "string",
    "ParentGeoLocationName": "string",
    "SpaceId": "string",
    "SpaceName": "string"
  },
  "Type": "string"
}

```

#### Response Status and Error Codes

Code	Description
200	Retrieved the Hierarchies for the Location of a Space successfully.
400	Invalid request. Wrong format or structure of the provided request.
404	Geolocation Space IDs not found.
500	Internal server error. The operation you requested led to an error during execution.

## 3.2.8 Authorization

The Geolocation services ensure that the services that are built on the platform are protected against unauthorized user access. You can secure the services by defining the services that an authenticated user can access, and also by defining the object instances that he or she can access.

Based on the authorization concept, an administrator grants permissions to the employees of an organization. These permissions can differ according to the tasks and responsibilities of a particular user. Granting permissions to users is realized by assigning users to different user groups, where the permissions that are related to a user group are automatically passed on to the members of that user group.

### 3.2.8.1 Role Template and Scopes

The Role-based authorization is an access control mechanism defined around roles and privileges. Roles are created for various job functions and the permissions to perform certain operations are assigned only to specific roles. Geolocation services offer a wide range of role templates as well as scopes, and providing (or revoke) functional authorization can become a significant challenge for administrators who are in charge of assigning proper user profiles for the users of the system. To make this task easier, Geolocation services provide a number of predefined role templates that an administrator can use as a blueprint for setting up individual roles, which are required for a concrete organization. The basic procedure for a role template is as follows:

1. Derive a role from a role template.
2. Assign role (one or many) to a user or user group.

Role templates and scopes have a n:m relationship. That is, one template may contain many scopes, and one scope may be contained in many templates. The following sections show listings of the available role templates and the scopes that are contained in each template. The listings are broken down by the key components of the Geolocation platform services.

## Geolocation Service

### Role Templates

Role Template	Task Description	Scopes
Geolocation_Viewer	View all resources in geolocation service	<geolocation>.r
Geolocation_Editor	Create and modify geolocation data by performing operations such as Read, Create, and Update.	<geolocation>.r <geolocation>.c <geolocation>.u
Geolocation_Super_Editor	Create, modify, or delete geolocation data by performing operations such as Read, Create, Update, and Delete.	<geolocation>.r <geolocation>.c <geolocation>.u <geolocation>.d

## 3.2.9 Error Response Structures

The error response structures for the APIs are as follows:

### Output Code

```
400 Bad Request
{
  "error": {
    "code": "400",
    "message": "Invalid resource payload."
  }
}
Descriptive messages to be used in some cases (e.g. deletion of a resource
with active references)
403 Deletion Forbidden
{
  "error": {
    "code": "403",
    "message": "Resource with ID XXX not found",
    "details": [
      {
        "code": "403",
        "target": "{RESOURCE URI}"
        "message": "Empty result set"
      }
    ]
  }
}
```

# 4 Authorization

Introduction into the authorization concept of the SAP Leonardo IoT platform

## Introduction

The authorization concept of the SAP Leonardo IoT platform ensures that the services that are built on the platform are protected against unauthorized access. This is done by defining the services that an authenticated user may access, and by defining the object instances that he or she may access.

Based on the authorization concept, an administrator grants permissions to the employees of an organization. These permissions may differ according to the tasks and responsibilities of a particular employee. Granting permissions to employees is realized by assigning employees to different user groups, where the permissions that are related to a user group are automatically passed on to the members of that user group.

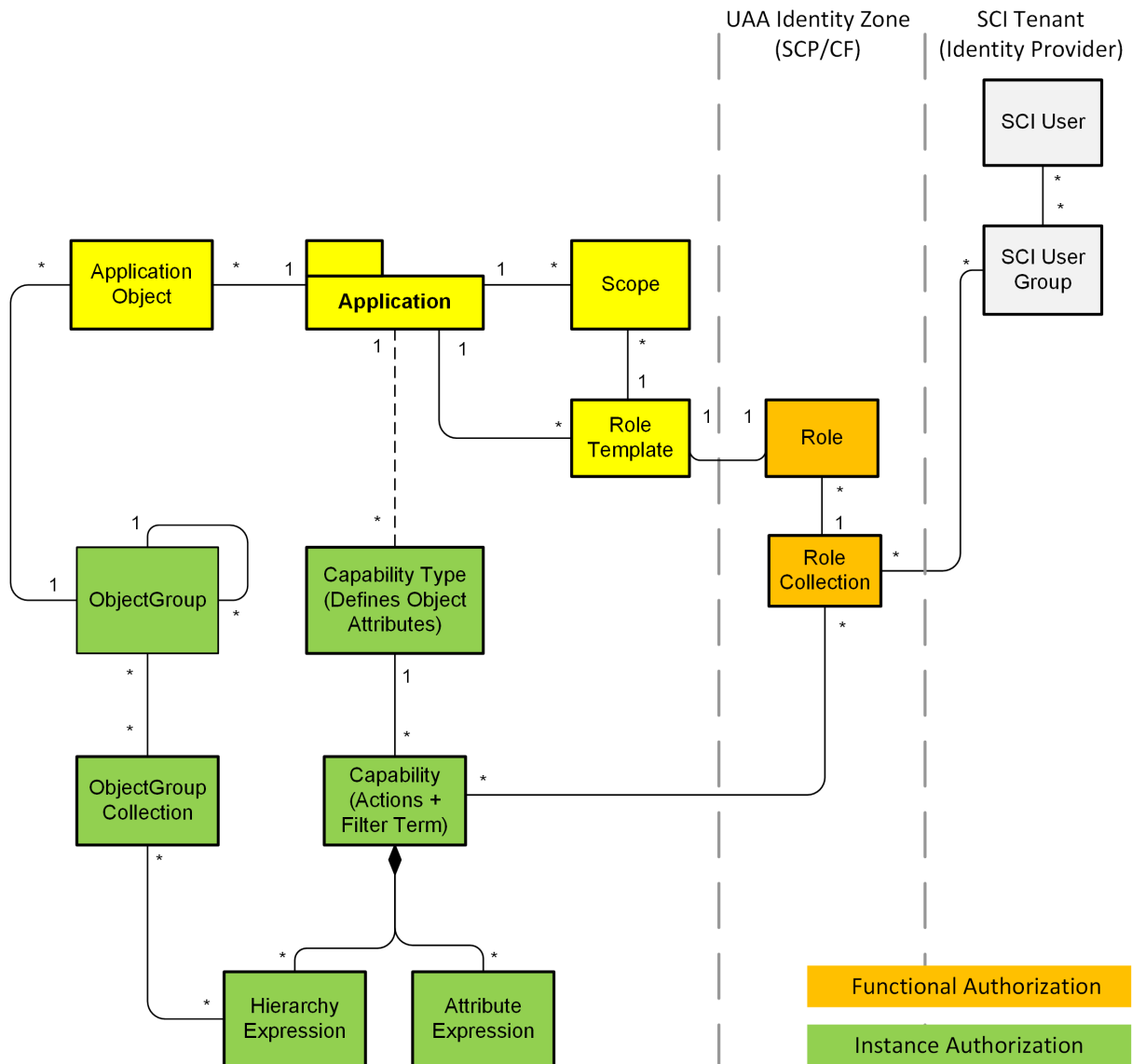
## Components of the Authorization Concept

The authorization concept of the SAP Leonardo IoT platform comprises the following components:

- You need an identity provider (IDP) service for creating users and user groups. For each tenant (that is, for each customer using the platform services), a separate IDP must be provided.
- **SAP User Account and Authentication Service (UAA):** This component is used as an authorization server. For each tenant, a dedicated UAA instance (sometimes referred to as identity zone) is provided. SAP UAA serves the purpose of assigning and checking of functional authorizations. These functional authorizations (also known as UAA scopes) control whether a particular user has the permission to start a given service. They are assigned to roles provided by the UAA service. For easier administration, the roles are then bundled to UAA role collections, which in turn are assigned to user groups provided by the identity provider.
- **IoT Business Partner Services:** With these onboarding services, it is easy to relate an employee of a platform customer with an identity provide user and to assign the employee to a user group.
- **IoT Authorization Services:** With these services, an administrator can define access rights for individual object instances. This kind of authorizations is used to filter the access to single instances of the application-specific objects managed in the SAP Leonardo IoT platform (for example, a particular welding robot installed at an assembly line of a car manufacturer plant in Munich, Germany). Application objects are assigned to authorization objects (represented by the ObjectGroup services offered by the platform). Object instance authorizations are directly assigned to user groups. As a prerequisite for calling application services, the user groups need the functional authorizations provided by UAA role collections. This concept results in a two-step process of checking whether a user has sufficient authorizations for a particular service request:
  1. Check whether the user is authorized for the desired activity (create/read/update/delete).
  2. If functional authorization is sufficient, check whether the user is authorized to perform the desired activity for a given object.

## Overview Diagram

The following diagram illustrates the entities involved in the authorization concept of the SAP Leonardo IoT platform:



### Note

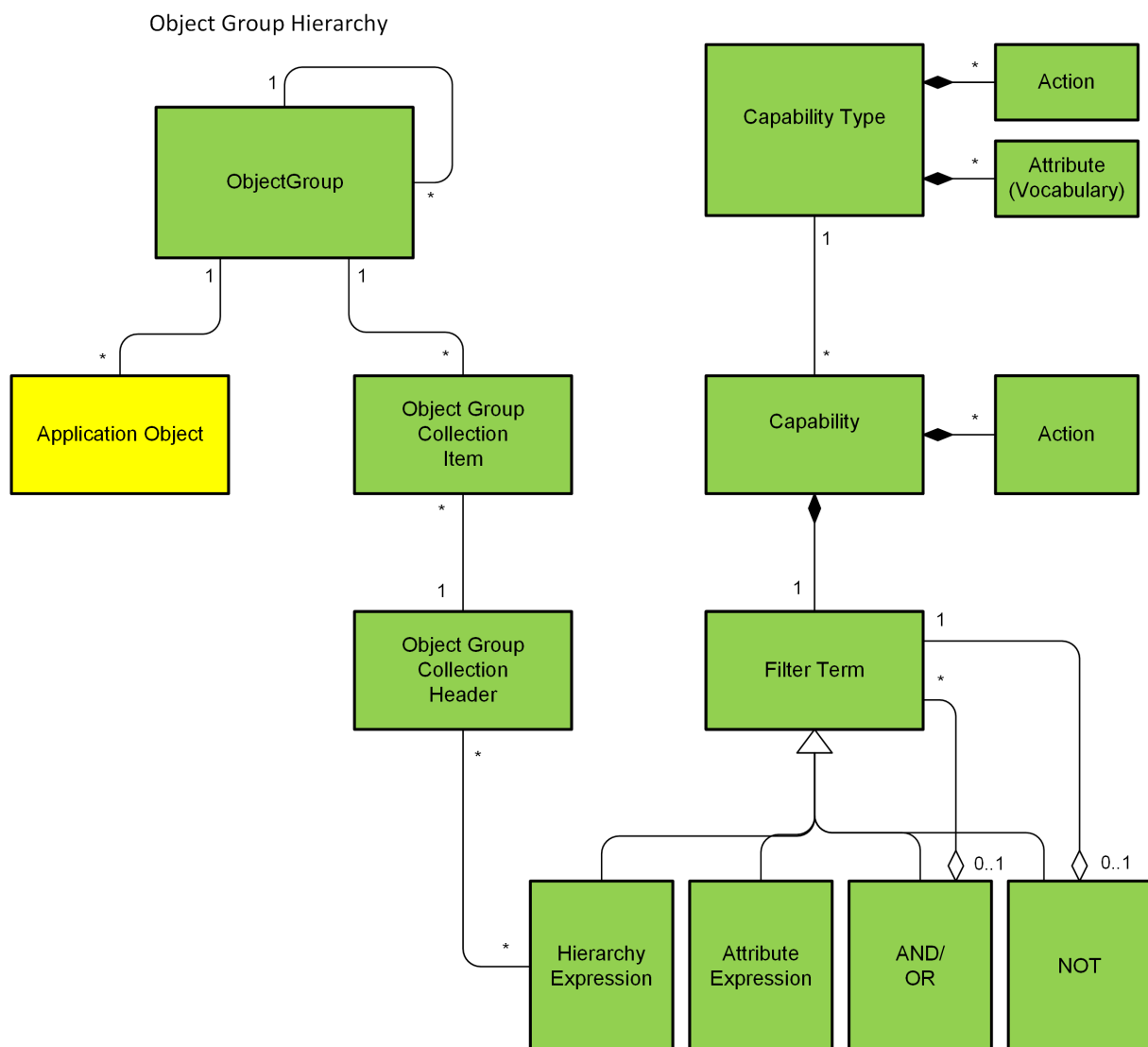
In the above diagram, the term "SCI" has been derived from SAP's identity provider service "SAP Cloud Identity". However, it should be understood as a placeholder for any other identity provider service you may have in use in your system landscape.

## IoT Authorization Services

An administrator uses the IoT Authorization Services to grant access authorizations on object instance level to user groups (and the members of these groups, respectively). Instance authorizations are based on a hierarchy of authorization groups. These authorization groups are represented by the `ObjectGroup` services offered by the platform. Applications running on the platform set up a relation between the application-specific objects and authorization groups to define who is allowed to access each object, and what kind of operations a user is allowed to perform with respect to an object.

The system administrator grants permission to read, write, or delete objects of a given type to a user group. With that, users are enabled to access object instances of that type. Access rights are inherited along the hierarchy tree of authorization groups. This inheritance is in effect for all subordinate nodes in the hierarchy as long as no new permission is either granted or revoked on a lower level. Once such a new permission has been defined, it remains effective for all subordinate nodes of that node, and so on.

The following diagram shows a detailed overview of the data model used for object instance-based authorizations:





## i Note

### General disclaimer:

The implementation of the authorization concept is subject to continuous refinement and adjustments to internal architectural revisions. Therefore, it is possible that the list of permissions documented for a particular service method comprises more scopes than actually needed for a particular use case at a particular point in time. However, to be always on the safe side, we strictly recommend to always provide all the scopes that are listed in the documentation. Otherwise, you might run into a situation where an application that used to run fine with a reduced set of authorizations fails in the future due to internal technical changes in the platform backend.

## Related Information

[Object Instance-based Authorization \[page 205\]](#)

[Functional Authorization \[page 263\]](#)

## 4.1 Object Instance-based Authorization

Services for managing object instance-based access rights

In this section, you find services for granting or revoking access rights to users who are logged on to the Leonardo IoT platform. The access rights in question here are assigned to specific objects of a certain type that are managed in the Leonardo IoT platform. Access rights are generally limited to objects within a particular tenant, thus ensuring that a user can only access the objects that belong to the same tenant to which he or she is assigned. The exception to this rule are users who are acting as administrators on the levels of a Cloud Platform Host or a Cloud Platform Provider (T0). For more information on the different levels, see [Business Partner \[page 25\]](#).

## Related Information

[ObjectGroup \[page 206\]](#)

[CapabilityType \[page 221\]](#)

[Capability \[page 225\]](#)

[Filter Terms \[page 236\]](#)

[ObjectGroupCollection \[page 241\]](#)

[ObjectGroupCollectionItem \[page 249\]](#)

## 4.1.1 ObjectGroup

An ObjectGroup is a technical object used for assigning object-based access authorizations to user groups.

An ObjectGroup is a technical object used for assigning object-based access authorizations (as opposed to function-based authorizations) to user groups. Capabilities for certain user groups and object types are assigned to object groups, granting read, write, or delete authorization for user groups to all objects that are assigned to this object group or to any child object group.

ObjectGroup entity nodes are organized in a mandatory hierarchical structure. This structure is used to control user access to certain object instances.

### Note

We highly recommend also reading the chapters dealing with the concept of the root object group. For more information, see [Root Object Group of a Tenant \[page 217\]](#).

#### Base URI:

- Formal description: `http://<server address>[:<port number>][<path>]/ObjectGroups`
- Example for a base URI in a cloud foundry environment:
- `https://sap-iotaeexplore.iot-sap.cfapps.eu10.hana.ondemand.com/authorization/ObjectGroups`

## Cross-Tenant Operations

A user acting as a Cloud Platform Consumer (T1) administrator can only access, create and delete object groups assigned to the tenant to which he or she is logged on. If the tenant is not provided in the payload of a *POST* or *PUT* request, it is derived from the logon tenant of the user. If the tenant is provided in the payload, but differs from the logon tenant, HTTP status code 403 ("Forbidden") is returned.

## Methods

HTTP Method	Action	URI	Scopes
<i>POST</i>	<a href="#">Create an ObjectGroup [page 207]</a>	<code>/ObjectGroups</code>	<code>&lt;auth&gt;.c</code>
<i>GET</i>	<a href="#">Read an ObjectGroup [page 209]</a>	<code>/ObjectGroups ('&lt;ID&gt;')</code>	<code>&lt;auth&gt;.r</code>
<i>GET</i>	<a href="#">Read all ObjectGroups [page 211]</a>	<code>/ObjectGroups</code>	<code>&lt;auth&gt;.r</code>
<i>PUT</i>	<a href="#">Update an ObjectGroup [page 214]</a>	<code>/ObjectGroups ('&lt;ID&gt;')</code>	<code>&lt;auth&gt;.u</code>
<i>DELETE</i>	<a href="#">Delete an ObjectGroup [page 216]</a>	<code>/ObjectGroups ('&lt;ID&gt;')</code>	<code>&lt;auth&gt;.d</code>

## Payload

Format: *JSON*

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "objectGroupID"      : { "type": "string", "minLength": 0, "maxLength": 32 },
  "objectGroupParentID": { "type": "string", "minLength": 32, "maxLength": 32,
    "required": "true" },
  "name"               : { "type": "string", "minLength": 1, "maxLength": 255,
    "required": "true" },
  "etag"               : { "type": "string" },
  "tenant"             : { "type": "string", "required": "true" }}
```

Field Explanation

Name	Description
objectGroupID	Unique ID of the ObjectGroup. <div><b>Note</b> Although this field is mandatory, its minimum length is still defined as 0. This is because the system ensures that the field always contains a proper value.</div>
objectGroupParentID	Unique ID of the superordinate ObjectGroup
name	Descriptive name of the ObjectGroup (for example, "High Pressure Water Pump")
etag	Entity tag used to identify a particular instance of an ObjectGroup
tenant	Name of the tenant to which an ObjectGroup is assigned

## Related Information

[UserGroup \[page 309\]](#)

[Root Object Group of a Tenant \[page 217\]](#)

### 4.1.1.1 Create an ObjectGroup

With this method, you create a new object group.

The system automatically assigns a unique ID to the new object group. It is therefore allowed to omit the `objectGroupID` field from the payload when this method is executed. Since each object group must refer to a superordinate parent object group, you must specify that parent object group by its ID.

The name of an object group may consist of lowercase or uppercase letters, digits, or underscores. It must have a length between 1 and 255 characters. Siblings in the object group hierarchy must not have the same name.

Also, you may decide that the new object group is assigned to a tenant different to the one where you are currently logged on. To accomplish this, you need to explicitly specify that tenant during creation of the object group. For more information on the prerequisites for this kind of operation, see the *Cross-Tenant Operations* section in [ObjectGroup \[page 206\]](#).

If the object group is successfully created, the server sends HTTP response code 201. Any other code indicates an error.

## Request

**URI:** /ObjectGroups

**HTTP Method:** *POST*

**Permissions:** <auth>.c

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token

### Request Fields

Name	Required	Type	Description	Send As
objectGroupParentID	Yes	string	ID of superordinate ObjectGroup	JSON
name	Yes	string	Semantic name of the ObjectGroup	JSON
tenant	No	string	If omitted, tenant of currently logged on user is used.	JSON

### Request Example

```
/ObjectGroups
```

## Payload

Format: *JSON*

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "objectGroupID"      : { "type": "string", "minLength": 0, "maxLength": 32 },
  "objectGroupParentID": { "type": "string", "minLength": 32, "maxLength": 32,
  "required": "true" },
```

```
"name" : { "type": "string", "minLength": 1, "maxLength": 255,
"required": "true" },
"etag" : { "type": "string" },
"tenant" : { "type": "string", "required": "true" }}
```

## Response

Format: *JSON*

### Response Header Fields

Name	Type	Description
Location	string	Path to the newly created ObjectGroup
Etag	string	entity tag of the newly created ObjectGroup instance

### Response Status and Error Codes

Code	Reason
201	ObjectGroup created successfully
403	Forbidden (user is not authorized to create an object group)
412	Precondition failed

## Related Information

[ObjectGroup](#) [page 206]

### 4.1.1.2 Read an ObjectGroup

Retrieves a specified object group from the database

With this method, you send a request to the server to retrieve a particular object group specified by its ID.

If the object group is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

## Request

**URI:** /ObjectGroups ('<objectGroupID>')

HTTP Method: [GET](#)

Permissions: <auth>.r

#### Request Fields

Name	Required	Type	Description	Send As
objectGroupID	Yes	string	ID of the ObjectGroup	Path parameter

#### Request Example

```
/ObjectGroups ('94c98d6a5fbe4aa0ad80d000667e4755')
```

## Response

Format: [JSON](#)

#### Response Header Fields

Name	Type	Description
Etag	string	Identifier of the retrieved ObjectGroup instance (applicable only in case of a single ObjectGroup)

#### Response Fields

Name	Type	Description	Send As
objectGroupID	string	ID of the ObjectGroup	JSON
objectGroupParentID	string	ID of superordinate ObjectGroup	JSON
name	string	Semantic name of the ObjectGroup	JSON
tenant	string	Tenant to which the ObjectGroup is assigned	JSON
etag	string	Entity tag of the returned ObjectGroup instance	JSON

#### Response Status and Error Codes

Code	Reason
200	ObjectGroup or ObjectGroups retrieved successfully
403	Forbidden (user is not authorized to read an object group)

Code	Reason
404	Not found (wrong objectGroupID)

## Payload

Format: *JSON*

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "objectGroupID"      : { "type": "string", "minLength": 0, "maxLength": 32 },
  "objectGroupParentID": { "type": "string", "minLength": 32, "maxLength": 32,
    "required": "true" },
  "name"               : { "type": "string", "minLength": 1, "maxLength": 255,
    "required": "true" },
  "etag"               : { "type": "string" },
  "tenant"             : { "type": "string", "required": "true" }}
```

## Related Information

[ObjectGroup \[page 206\]](#)

### 4.1.1.3 Read all ObjectGroups

Retrieves a list of object groups from the database

With this method, you send a request to the server to retrieve a subset of all object groups, according to the filter criteria provided. For a collective request for a set of object groups, you have these options:

- You can sort and restrict the set of object groups to be retrieved based on the following fields:

- tenant
- objectGroupID
- objectGroupParentID
- name

Defining filter criteria is **not** mandatory, but recommended because of the typically high number of object group instances.

- The `ObjectGroup` service offers an additional set of filter parameters that allows for retrieving object groups belonging to a certain hierarchy path. This hierarchy-based filter cannot be combined with any other filter criteria. Object groups matching the hierarchy filter criterion are returned as a simple, unsorted list of items that does not offer any chance of restoring the hierarchical relationships between the object groups returned. The object group used as the starting point is **not** part of the returned result. With an additional `MaxDistance` parameter, you can control the number of hierarchy levels that shall be included in the response. Using this parameter is optional. If it is omitted, the entire hierarchy based on the given starting point will be processed.

- You can define that the set of matching object groups shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

You can retrieve the global tenant root object group by calling this service with the following filter criteria:

```
objectGroupParentID eq null
```

If the set of object groups is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

### Note

Retrieving high numbers of objects from the database with only one collective GET request can lead to high system load and a significant increase of response times. Therefore, we highly recommend limiting the maximum number of returned objects to no more than 500 per method call. This can easily be accomplished with the help of the `$top` and `$skip` query parameters.

To avoid unwanted performance decline, collective GET requests are only supported up to a maximum of 500 objects that can be retrieved with one single call. Therefore, in use cases where it is crucial for you to ensure that all existing objects have been retrieved, you need to verify this by setting up a loop construct in your source code to repeat the request until the service method returns no further objects.

## Request

**URI:** `/ObjectGroups`

**HTTP Method:** `GET`

**Permissions:** `<auth>.r`

### Query String Parameters

Name	Required	Type	Description
<code>\$skip</code>	No	integer	Number of the first n records to be excluded from the result set
<code>\$top</code>	No	integer	Number of records to include in the result set
<code>\$orderby</code>	No	string	Field name to be used for sorting the result set.
<code>\$filter</code>	No	string	Filter condition to be applied; valid fields are <code>tenant</code> , <code>name</code> , <code>objectGroupID</code> , and <code>objectGroupParentID</code>  Hierarchy filter: Special syntax required, see example below.

### Request Example

```
/ObjectGroups
```



Retrieves all object groups that are available in the database. Note that this unrestricted request is not recommended due to high system load.

```
/ObjectGroups?$filter=objectGroupParentID eq null
```

Retrieves all object groups that are defined within the scope of the root object group. Capabilities that are assigned to object groups on this hierarchy level are inherited by lower level object groups as long as there is no concurrent capability assignment made, which would overwrite the root assignments.

```
/ObjectGroups?$top=25&$filter=objectGroupParentID eq  
'94c98d6a5fbe4aa0ad80d000667e4755'&$orderby=name
```

Retrieves the first 25 object groups that are subordinate to the parent object group with the given ID. Data is retrieved in page mode to reduce network load and response times. The list of object groups is sorted by their name in ascending order.

```
/ObjectGroups?$skip=10&$top=20
```

Retrieves 20 object groups from the database where the first 10 object groups are omitted. That is, the retrieved object groups are those with sequence numbers 11 through 30.

```
/ObjectGroups?$filter=$it/  
Aggregation.isdescendant(Hierarchy='ObjectGroup',Node='94c98d6a5fbe4aa0ad80d00066  
7e4755',maxDistance=2),
```

Hierarchy-based filter: Returns a list of object groups that have been defined as sub-object groups of the object group with the specified ID. The result is limited to the first and second hierarchy level below the specified object group.

## Response

### Response Status and Error Codes

Code	Reason
200	List of ObjectGroups retrieved successfully
400	Bad request (for example, invalid or missing filter)
403	Forbidden (user not authorized to read object groups)

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method has the following structure:

```
{  
  "objectGroupID" : { "type": "string", "minLength": 0, "maxLength": 32 },  
}
```

```

"objectGroupParentID" : { "type": "string", "minLength": 32, "maxLength": 32,
"required": "true" },
"name"                : { "type": "string", "minLength": 1, "maxLength": 255,
"required": "true" },
"etag"                : { "type": "string" },
"tenant"              : { "type": "string", "required": "true" }}

```

## Related Information

[ObjectGroup \[page 206\]](#)

### 4.1.1.4 Update an ObjectGroup

With this method, you modify an object group with the specified ID.

In the payload of the request, you assign values to those fields that you want to modify. For example, to attach the specified object group to a new superordinate object group, you pass the ID of the new superordinate object group via the `objectGroupParentID` parameter.

#### Note

You cannot change the ID of an object group. The ID is automatically generated by the system upon creation of an object group and must remain stable throughout its lifecycle.

If the object group is successfully updated, the server sends HTTP response code 200. Any other code indicates an error.

## Request

**URI:** `/ObjectGroups (<objectGroupID>)`

**HTTP Method:** *PUT*

**Permissions:** `<auth>.u`

#### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross site request forgery token
etag	No	Identifier of ObjectGroup instance version
If-Match	No	Check if cached version matches server version of the ObjectGroup

#### Request Fields

Name	Required	Type	Description	Send As
objectGroupID	Yes	string	ID of the ObjectGroup	Path parameter
objectGroupParentID	Yes	string	ID of superordinate ObjectGroup	JSON
name	Yes	string	Semantic name of the ObjectGroup	JSON
tenant	No	string	If omitted, tenant of currently logged on user is used.	JSON

### Request Example

```
/ObjectGroups ('94c98d6a5fbe4aa0ad80d000667e4755')
```

Updates the fields of the specified object group with the values provided with the payload of the request.

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "objectGroupID"      : { "type": "string", "minLength": 0, "maxLength": 32 },
  "objectGroupParentID": { "type": "string", "minLength": 32, "maxLength": 32,
  "required": "true" },
  "name"               : { "type": "string", "minLength": 1, "maxLength": 255,
  "required": "true" },
  "etag"               : { "type": "string" },
  "tenant"             : { "type": "string", "required": "true" }}
```

## Response

Format: [JSON](#)

### Response Header Fields

Name	Type	Description
Etag	string	Identifier of the updated ObjectGroup instance version

### Response Status and Error Codes

Code	Reason
200	ObjectGroup updated successfully

Code	Reason
403	Forbidden
404	Not found
409	Conflict (for example, object instance has been changed)
412	Precondition failed

## Related Information

[ObjectGroup \[page 206\]](#)

### 4.1.1.5 Delete an ObjectGroup

Deletes a specified object group.

With this method, you physically delete the object group with the specified ID from the database.

Deleting of an object group is **not** possible if objects are assigned to the object group or if the object group has children in the object group hierarchy. If an object group is deleted, all assigned object group capabilities are deleted as well.

If the object group is successfully deleted, the server sends HTTP response code 204. Any other code indicates an error.

## Request

**URI:** `/ObjectGroups ('<objectGroupID>')`

**HTTP Method:** `DELETE`

**Permissions:** `<auth>.d`

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross site request forgery token
If-Match	Yes	Check if cached version matches server version of the ObjectGroup

### Request Fields

Name	Required	Type	Description	Send As
objectGroupID	Yes	string	ID of the ObjectGroup	Path parameter

### Request Example

```
/ObjectGroups ('94c98d6a5fbe4aa0ad80d000667e4755')
```

Deletes the object group with the specified ID.

## Response

### Response Status and Error Codes

Code	Reason
204	ObjectGroup deleted successfully

## Related Information

[ObjectGroup \[page 206\]](#)

## 4.1.2 Root Object Group of a Tenant

This service helps to determine the root ObjectGroup of a tenant.

In each tenant, there is exactly one ObjectGroup instance that serves as the entry point into the object group hierarchy defined for that tenant. This hierarchy root entry stands out of all other ObjectGroup instances by the fact that this is the only instance within a tenant where the `objectGroupParentID` field is empty (although that field is defined as mandatory for all other entries found in the ObjectGroup hierarchy; however, this mandatory requirement is internally overridden for the root ObjectGroup). However, browsing through the list of all ObjectGroup instances that exist in a tenant can be tedious and impose unnecessary load on the system. The `ObjectGroups/TenantRoot` service is a specialized service with the single purpose of determining that particular ObjectGroup instance with one single method call.

## Root Object Group Determination

The different scenarios that are covered by this method depend on various preconditions according to the following table:

Root Object Group Determination

Tenant Level	Tenant Onboarding Status	
	Legacy	New Onboarding
T1	<code>objectGroupParentID = &lt;cross-tenant root object group in T0&gt;</code>	<code>objectGroupParentID = Null</code>

### → Recommendation

With respect to the different behavior of the `objectGroupParentID` field, we highly recommend always using the root object group service described here for determining the root object group. The service automatically detects whether a tenant is a legacy tenant or a tenant that has been created via the new onboarding strategy. The service then automatically adapts to the nature of the tenant in question and returns the correct root object group.

### Base URI:

- Formal description: `http://<server address>[:<port number>][/path]/ObjectGroups/TenantRoot`
- Example for a base URI in a cloud foundry environment: `https://sap-iotaexplore.iot-sap.cfapps.eu10.hana.ondemand.com/appiot-mds/ObjectGroups/TenantRoot`

## Methods

Due to the highly specialized purpose of this service, only the GET method is supported. Once the root object group has been determined, you can execute the various methods provided by the ObjectGroup service. This, however, should only be done with extra care because of the important role of the root object group.

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read the Root ObjectGroup [page 219]</a>	<code>/ObjectGroups/TenantRoot</code>	<code>&lt;auth&gt;.r</code>

## Payload

Since the root ObjectGroup has the same structure as any other ObjectGroup instance, its payload is exactly the same as the one of all other object groups:

```
{
  "objectGroupID"      : { "type": "string", "minLength": 0, "maxLength": 32 },
  "objectGroupParentID": { "type": "string", "minLength": 32, "maxLength": 32,
  "required": "true" },
}
```

```

    "name" : { "type": "string", "minLength": 1, "maxLength": 255,
    "required": "true" },
    "etag" : { "type": "string" },
    "tenant" : { "type": "string", "required": "true" }}

```

## Related Information

[ObjectGroup \[page 206\]](#)

### 4.1.2.1 Read the Root ObjectGroup

Returns the root object group of a tenant

With this method, you can determine the root object group of either the current tenant or of a tenant specified via parameter.

#### Root Object Group Determination

The different scenarios that are covered by this method depend on various preconditions according to the following table:

Root Object Group Determination

Tenant Level	Tenant Onboarding Status	
	Legacy	New Onboarding
T1	objectGroupParentID = <cross-tenant root object group in T0>	objectGroupParentID = Null

#### → Recommendation

With respect to the different behavior of the `objectGroupParentID` field, we highly recommend always using the root object group service described here for determining the root object group. The service automatically detects whether a tenant is a legacy tenant or a tenant that has been created via the new onboarding strategy. The service then automatically adapts to the nature of the tenant in question and returns the correct root object group.

For an explanation of the different tenant levels, see section *Business Partner in SAP Leonardo IoT* in the [Business Partner \[page 25\]](#) chapter.

## Request

**URI:** `/ObjectGroups/TenantRoot[?tenant=<tenant ID>]`

**HTTP Method:** [GET](#)

Permissions: <auth>.r

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
tenant	No	string	ID of the desired tenant. Can only be used for referring to T1 tenants when logged on to a T0 tenant. Defaults to the currently logged on tenant if omitted.	Query

## Request Example

```
/ObjectGroups/TenantRoot?tenant=wdf-customer15
```

## Response

### Response Example

```
{
  "objectGroupID"      : "9764513e0fd540f09108e365d82f947c",
  "name"               : "car500wheel_fl",
  "etag"               : "3",
  "tenant"             : "wdf-customer15"
```

Note that in the above example, the `objectGroupParentID` field is omitted because it contains a Null value, according to the new onboarding strategy.

### Response Status and Error Codes

Code	Reason
200	Root ObjectGroup retrieved successfully
403	Forbidden (user is not authorized to read the root object group)
404	Not found (wrong objectGroupID)

## Related Information

[Root Object Group of a Tenant \[page 217\]](#)



## 4.1.3 CapabilityType

A `CapabilityType` serves as a template for a `Capability`.

`CapabilityType` objects provide the definition of a certain capability that can be granted to a user with respect to certain objects. The relation between a `CapabilityType` and a `Capability` is similar to the relation between a class and an object instance in object-oriented programming languages. That is, the essential properties of a given `Capability` are defined by the `CapabilityType` from which it is derived.

In SAP Leonardo IoT, `CapabilityTypes` are treated as a very low-level kind of business content. That is, there are hard-coded, predefined `CapabilityTypes` available that are delivered by SAP and that can neither be changed nor extended by customers. However, the number of `CapabilityTypes` may grow in future releases. The following `CapabilityTypes` are currently available:

- Thing
- Organization
- Person

### Base URI:

- Formal description: `http://<server address>[:<port number>][<path>]/CapabilityTypes`
- Example for a base URI in a cloud foundry environment: `http://authorization.cfapps.neo.ondemand.com/CapabilityTypes`

**Permissions:** `<auth>.r`

## Methods

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read a CapabilityType [page 223]</a>	<code>/CapabilityTypes ('&lt;ctID&gt;')</code>	<code>&lt;auth&gt;.r</code>
GET	<a href="#">Read all CapabilityTypes [page 224]</a>	<code>/CapabilityTypes</code>	<code>&lt;auth&gt;.r</code>

## Structure

Unlike almost all of the other services in SAP Leonardo IoT, `CapabilityTypes` do **not** appear in the persistence layer. This is due to their hardcoded nature, which leads to a mere transient usage with no representation in the database.

However, the hardcoded internal structure of the service can of course be described. It consists of the following parts:

#### Field Explanation

Name	Description
ID	<p>Unique identifier for the capability type. It matches the following pattern: <code>FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF</code>. Here, the asterisks used as placeholder elements ("*") are to be replaced by a unique hexadecimal code, and they are reserved for capability types shipped by SAP. The values for these predefined capability types are as follows:</p> <ul style="list-style-type: none"><li>• Person: <code>FFFFFFFFF0000000FFFFFFFFF0001</code></li><li>• Organization: <code>FFFFFFFFF0000000FFFFFFFFF0002</code></li><li>• Thing: <code>FFFFFFFFF0000000FFFFFFFFF0003</code></li></ul>
Name	<p>Semantic identifier for the capability type. These names are:</p> <ul style="list-style-type: none"><li>• <code>thing:thing</code></li><li>• <code>bp:organization</code></li><li>• <code>bp:person</code></li></ul>
Actions	<p>A set of actions that are available for a particular type. Currently the following actions are supported:</p> <ul style="list-style-type: none"><li>• Read</li><li>• Write</li><li>• Delete</li></ul>
Vocabulary	<p>The set of available fields and their respective database entity. Only fields with a 1:1 relationship are allowed. The main source is the respective <code>Databasetable</code> or <code>View</code> underlying the "Get all" REST service. The <code>Vocabulary</code> has its own package where the respective classes with all needed methods are implemented.</p>

With the field structure given above, the payload structure of `CapabilityType` service varies depending on the particular type in question. Here, not only the predefined field values for ID and Name are different. In addition, the structure of the `Vocabulary` field is type-specific and results in a very complex payload structure. Because of that, we have outsourced the payload into three separate topics. For more information, see the [Related Information](#) section at the bottom of this topic.

## Related Information

[Capability](#) [page 225]

[Payload of CapabilityType "Person"](#) [page 73]

[Payload of CapabilityType "Organization"](#) [page 44]

[Payload of CapabilityType "Thing"](#) [page 746]

### 4.1.3.1 Read a CapabilityType

With this method, you can check the existence of a particular `CapabilityType`

You can use this method to find out whether a particular `CapabilityType` is available in the system landscape or not. This can be helpful when you experience technical problems and want to check whether the system is in a healthy state. Since the `CapabilityType` service has no persistence layer, there is no other use case for this method.

#### Request

**URI:** `/CapabilityTypes('<ID>')`

**HTTP Method:** `GET`

**Permissions:** `<auth>.r`

#### Query String Parameters

Parameter	Required	Data Type	Description	Parameter Type
ID	Yes	string	Unique identifier of the <code>CapabilityType</code>	Path

#### Request Example

```
/CapabilityTypes('FFFFFFFFF00000000FFFFFFFFF0001')
```

#### Response

##### Response Status and Error Codes

Code	Description
200	<code>CapabilityType</code> retrieved successfully

#### Related Information

[CapabilityType \[page 221\]](#)

## 4.1.3.2 Read all CapabilityTypes

Retrieves a list of all `CapabilityType` objects from the database

You use this method to find out which `CapabilityTypes` are available in the system and can be used for deriving individual capabilities for certain objects.

### Request

**URI:** `/CapabilityTypes`

**HTTP Method:** `GET`

**Permissions:** `<auth>.r`

### Request Parameters

None.

#### i Note

Since the number of predefined `CapabilityType` objects is very small, there is no need for any kind of filtering or restricting parameters that are otherwise common for the "Get all" methods of most of the other services.

### Request Example

```
/CapabilityTypes
```

### Response

#### Response Status and Error Codes

Code	Description
200	CapabilityTypes retrieved successfully

### Related Information

[CapabilityType \[page 221\]](#)

## 4.1.4 Capability

An object instance of a `CapabilityType` that can be associated with a role collection.

You use `Capability` objects to specify a set of objects in the database on which users can perform actions as defined by the `CapabilityType` from which the `Capability` has been derived. The multiple dimensions of authorization that are involved here can be grouped as follows:

Authorization Dimensions

Dimension	Explanation
Object set	Set of objects that a user can access defined by a complex filter expression given as <code>filterTerm</code> . The filter can be based on specific field values of a particular object such that granting the authorization is based on the presence (or absence) of a particular value.
Role collection	A set of roles of which at least one a user must be assigned to in order to be able to perform the actions controlled by the <code>Capability</code> .
Actions	The different actions (read, write, delete) that the user can perform on the objects defined by the object set.

The most powerful feature here is the definition of a set of objects for which access is granted. Here, a complex filter expression can be defined, which allows to restrict access not only to objects of a particular type, but to objects of a particular type with a particular value in one of its fields (or with different values in a list of its fields). This lets you define access rights on a very fine level of granularity, which in turn is a prerequisite for a clear segregation of duties needed to fulfill the data protection requirements for a multitenancy platform.

### ❖ Example

You want to define an authorization for a group of users who are in charge of monitoring a number of physical devices in a factory located in Munich (Germany). To accomplish this, you define a `Capability` object and assign a role collection for the user group in question where the set of accessible objects is restricted to those of the given type in question (for example, a welding robot) and that, at the same time, have the `cityName` field of the associated `Location` object set to 'Munich'.

### Cross-Tenant Maintenance:

Regarding cross-tenant maintenance, the following applies:

- As opposed to older releases of SAP Leonardo IoT, cross-tenant maintenance is possible only for cloud platform providers of SAP Leonardo IoT.
- Users that are assigned to a tenant of a cloud platform consumer (T1) can only access `Capability` objects that belong to the tenant where they are logged on to.

### Multiple Capabilities per User:

If a user has more than one `Capability` objects assigned that refer to the same set of objects, the access rights defined in the different `Capability` objects are combined via a logical OR, that is, the effective permission is a superset of all permissions granted by the `Capability` objects.

### Base URI:

- Formal description: `http://<server address>[:<port number>][<path>]/Capabilities`
- Example for a base URI in a cloud foundry environment: `https://authorization.cfapps.eu10.hana.ondemand.com/Capabilities`

## Methods

HTTP Method	Action	URI	Scopes
<i>POST</i>	<a href="#">Create a Capability [page 227]</a>	/Capabilities	<auth>.c
<i>GET</i>	<a href="#">Read a Capability [page 230]</a>	/Capabilities('ID')	<auth>.r
<i>GET</i>	<a href="#">Read all Capabilities [page 231]</a>	/Capabilities	<auth>.r
<i>PUT</i>	<a href="#">Update a Capability [page 234]</a>	/Capabilities('ID')	<auth>.u
<i>DELETE</i>	<a href="#">Delete a Capability [page 235]</a>	/Capabilities('ID')	<auth>.d

## Payload

Only media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "ID" : { "type": "string", "maxLength": 32 },
  "capabilityTypeID" : { "type": "string", "maxLength": 32, "required":
"true" },
  "name" : { "type": "string", "maxLength": 255, "required":
"true" },
  "userGroupTenant" : { "type": "string", "maxLength": 36 },
  "filterTerm" : { "type": "json" },
  "etag" : { "type": "string" },
  "userGroups" : { "value": {
255 } ]
    [ "userGroupName" : { "type" : "string", "maxLength":
    } },
    "actions" : { "value": {
255 } ]
      [ "actionName" : { "type" : "string", "maxLength":
      } }
    }
}
```

### i Note

- The name of a `Capability` object must be unique within a tenant.
- The `filterTerm` attribute declaration as `"type": "json"` in the payload listed above indicates that this field must be filled with a valid object selection criterion according to the JSON notation that is defined for filter terms. For more information and an example, see [Filter Terms \[page 236\]](#).

## Related Information

[Filter Terms \[page 236\]](#)

[CapabilityType \[page 221\]](#)

[UserGroup \[page 309\]](#)

## 4.1.4.1 Create a Capability

With this method, you create a new capability. The system automatically assigns a unique ID to the new capability.

Creating a new capability requires that you provide values for the following mandatory fields:

- `capabilityTypeID`

### i Note

You can retrieve the available values for this field with the [Read all CapabilityTypes \[page 224\]](#) method.

- `name`

If no value is given for the `userGroupTenant`, the system assigns the tenant of the currently logged on user to the new capability.

### i Note

If no value is given for the `filterTerm` field, the system raises an error message to make you aware of this. In this case, a new capability cannot be created because the `filterTerm` defines the set of objects for which the capability shall grant access permission. Providing a value for this field is essential and cannot be omitted.

## Request

**URI:** `/Capabilities`

**HTTP Method:** `POST`

**Permissions:** `<auth>.c`

## Request Parameters

None.

## Payload

Format: `JSON`

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "ID" : { "type": "string", "maxLength": 32 },
  "capabilityTypeID" : { "type": "string", "maxLength": 32, "required":
"true" },
  "name" : { "type": "string", "maxLength": 255, "required":
"true" },
  "userGroupTenant" : { "type": "string", "maxLength": 36 },
  "filterTerm" : { "type": "json" },
  "etag" : { "type": "string" },
  "userGroups" : { "value": {
    [ "userGroupName" : { "type" : "string", "maxLength":
255 } ]
  } },
}
```

```

    "actions"          : { "value": {
255 } ]               [ "actionName" : { "type" : "string", "maxLength":
                        } }
  }

```

## Request Example

/Capabilities

Here is a pretty simple example of a payload for a request where a new capability is created that grants full access (read, write, and delete) to all objects of type thing without any restriction criteria. The trick to accomplish this is to make use of the special value assignment "filterTerm": true instead of a highly differentiated and complex filterTerm statement that follows in the second example further below. But let's start with the simple version:

```

{
  "capabilityTypeID": "FFFFFFFFF00000000FFFFFFFFF0003",
  "name"            : "<name of your capability type>",
  "filterTerm"      : true,
  "etag"            : "1",
  "userGroups": {
    "value": [
      {
        "userGroup": "<your user group>"
      }
    ]
  },
  "actions": {
    "value": [
      {
        "action": "Write"
      },
      {
        "action": "Read"
      },
      {
        "action": "Delete"
      }
    ]
  }
}

```

### Note

The simple filterTerm condition used in the example above corresponds to the [Unrestricted](#) checkbox that you can find in the [Create Capability](#) and [Edit Capability](#) dialog box of the Tenant Administration app.

Here is a more complex example of a payload for a request where a new capability is created that grants full access (read, write, and delete) to an object of type organization matching the following criteria:

- cityName = "Berlin" AND objectGroupCollection = "OGC\_ID\_1000" OR
- cityName = "Heidelberg" AND objectGroupCollection = "OGC\_ID\_2000"

The new capability shall be assigned to a user group called "UserGroup\_3000", so that all users assigned to that user group have access to Organization objects matching the specified criteria:

```

{
  "capabilityTypeID": "FFFFFFFFF00000000FFFFFFFFF0002",
  "name": "{CapaName}",
  "filterTerm":

```



```

{
  "type": "FilterTermExpression",
  "booleanOperator": "OR",
  "filterTerms": [{
    "type": "FilterTermExpression",
    "booleanOperator": "AND",
    "filterTerms": [{
      "type": "FilterTermElement",
      "path": {
        "field": "cityName",
        "dimension": "root"
      },
      "operator": "EQUAL",
      "value": "Berlin"
    },
    {
      "type": "FilterTermHierarchyAssignment",
      "path": {
        "field": "objectGroup",
        "dimension": "root"
      },
      "objectGroupCollections": ["{{OGC_ID_1000}}"]
    }
  ]
},
{
  "type": "FilterTermExpression",
  "booleanOperator": "AND",
  "filterTerms": [{
    "type": "FilterTermElement",
    "path": {
      "field": "cityName",
      "dimension": "root"
    },
    "operator": "EQUAL",
    "value": "Heidelberg"
  },
  {
    "type": "FilterTermHierarchyAssignment",
    "path": {
      "field": "objectGroup",
      "dimension": "root"
    },
    "objectGroupCollections": ["{{OGC_ID_2000}}"]
  }
]
},
{
  "userGroups": {
    "value": [{
      "userGroup": "{{UserGroup_3000}}"
    }]
  },
  "actions": {
    "value": [
      {
        "action": "Read"
      },
      {
        "action": "Write"
      },
      {
        "action": "Delete"
      }
    ]
  }
}

```

## Response

### Response Status and Error Codes

Code	Description
201	Object created successfully

## Related Information

[Capability](#) [page 225]

[Filter Terms](#) [page 236]

## 4.1.4.2 Read a Capability

Retrieves the specified `Capability` from the database

With this method, you send a request to the server to retrieve a particular `Capability` object specified by its ID.

### Note

With a user assigned to a tenant on T1 level (Cloud Platform Consumer), you can only access `Capability` objects that belong to the currently logged on user.

## Request

URI: `/Capabilities('<ID>')`

HTTP Method: `GET`

Permissions: `<auth>.r`

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
ID	Yes	string	Unique identifier of the Capability	Path

### Request Example

```
/Capabilities('be531d4cfd524f06b8df7c999377fac8')
```

## Response

### Payload

```
{
  "ID" : { "type": "string", "maxLength": 32 },
  "capabilityTypeID" : { "type": "string", "maxLength": 32, "required":
"true" },
  "name" : { "type": "string", "maxLength": 255, "required":
"true" },
  "userGroupTenant" : { "type": "string", "maxLength": 36 },
  "filterTerm" : { "type": "json" },
  "etag" : { "type": "string" },
  "userGroups" : { "value": {
    [ "userGroupName" : { "type" : "string", "maxLength":
255 } ]
  } },
  "actions" : { "value": {
    [ "actionName" : { "type" : "string", "maxLength":
255 } ]
  } }
}
```

### Response Status and Error Codes

Code	Description
200	Capability retrieved successfully

## Related Information

[Capability \[page 225\]](#)

### 4.1.4.3 Read all Capabilities

Retrieves a list of `Capability` objects from the database

With this method, you send a request to the server to retrieve a subset of all `Capability` objects, according to the filter criteria provided. For a collective request for a set of `Capability` objects, you have these options:

- You can sort and restrict the set of `Capability` objects to be retrieved by filter criteria based on the following fields:
  - ID
  - `capabilityTypeID`
  - name
  - `userGroupTenant`
  - `crossTenantScope`
  - `actions/name`
  - `userGroups/userGroup`

- You can define that the set of matching `Capability` objects shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

### i Note

Retrieving high numbers of objects from the database with only one collective GET request can lead to high system load and a significant increase of response times. Therefore, we highly recommend limiting the maximum number of returned objects to no more than 500 per method call. This can easily be accomplished with the help of the `$top` and `$skip` query parameters.

To avoid unwanted performance decline, collective GET requests are only supported up to a maximum of 500 objects that can be retrieved with one single call. Therefore, in use cases where it is crucial for you to ensure that all existing objects have been retrieved, you need to verify this by setting up a loop construct in your source code to repeat the request until the service method returns no further objects.

## Request

URI: `/Capabilities`

HTTP Method: `GET`

Permissions: `<auth>.r`

### Query String Parameters

Name	Required	Type	Description
<code>\$skip</code>	No	integer	Number of the first n records to be excluded from the result set
<code>\$top</code>	No	integer	Number of records to include in the result set
<code>\$orderby</code>	No	string	Field name to be used for sorting the result set in ascending or descending order
<code>\$filter</code>	No	string	Filter condition to be applied; for valid fields, see list given above.

### i Note

For the filter parameter, the following applies:

- Defining a filter is **not** mandatory for this service method.
- The filter behavior is different depending on the tenant level of the currently logged on user:
  - User is assigned to a T0 tenant (Cloud Platform Provider): Defining a filter definition can be used to restrict the result set to those capabilities belonging to the specified tenant. Otherwise, all `Capability` objects for all tenants are returned.
  - User is assigned to a T1 tenant (Cloud Platform Consumer): Only the `Capability` objects belonging to the user's tenant are returned. If a different tenant is specified in the filter, the system returns an empty result list.

## Request Parameters

None.

## Request Example

```
/Capabilities
```

Retrieves all `Capability` objects that are available for the currently logged on user, or all `Capability` objects in case the user is assigned to a TO tenant **and** has been granted cross-tenant maintenance authorizations.

```
/Capabilities?$filter=userGroups/userGroup eq 'BP-FA-Group'
```

Retrieves all capabilities that are granted to users who are assigned to the specified user group.

```
/Capabilities?$filter=actions/name eq 'Read' and userGroups/userGroup eq 'BP-FA-Group'
```

Retrieves all capabilities that grant read access to users who are assigned to the specified user group. For specifying an action, you can use the string literals 'Read', 'Write', or 'Delete'.

```
/Capabilities?$top=25
```

Retrieves the first 25 `Capability` objects from the database.

```
/Capabilities?$filter=userGroupTenant eq 'customer-test001'
```

Retrieves all `Capability` objects that refer to the specified `userGroupTenant`.

```
/Capabilities?$skip=10&$top=20
```

Retrieves 20 `Capability` objects from the database where the first 10 objects are omitted. That is, the retrieved objects are those with sequence numbers 11 through 30.

## Response

### Response Status and Error Codes

Code	Description
200	Objects retrieved successfully

## Related Information

[Capability \[page 225\]](#)

## 4.1.4.4 Update a Capability

With this method, you modify a `Capability` object with the specified ID

In the payload of the request, you assign values to those fields that you want to modify. For example, assume you have a `Capability` defined that grants access rights to a group of users for a number of machines (modeled as `Thing` objects), where the machines are selected by the location data of their technical place. Now, a decision has been made to relocate the machines to a different assembly line in the same plant. You could then decide to update the respective `Capability` object by adjusting the selection criterion that is part of the `FilterTerm` field value rather than setting up completely new capabilities.

### Request

**URI:** `/Capabilities('<ID>')`

**HTTP Method:** *PUT*

**Permissions:** `<auth>.u`

### Request Header Fields

Header	Required	Description
If-Match	Yes	Check if cached version matches server version of the capability

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
ID	Yes	string	Unique identifier of the <code>Capability</code> object	Path

### Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "ID" : { "type": "string", "maxLength": 32 },
  "capabilityTypeID" : { "type": "string", "maxLength": 32, "required":
"true" },
  "name" : { "type": "string", "maxLength": 255, "required":
"true" },
  "userGroupTenant" : { "type": "string", "maxLength": 36 },
  "filterTerm" : { "type": "json" },
  "etag" : { "type": "string" },
  "userGroups" : { "value": {
    [ "userGroupName" : { "type" : "string", "maxLength":
255 } ]
  } },
  "actions" : { "value": {
    [ "actionName" : { "type" : "string", "maxLength":
255 } ]
  } }
}
```

## Request Example

```
/Capabilities('be531d4cfd524f06b8df7c999377fac8')
```

## Response

### Response Status and Error Codes

Code	Description
200	Capability updated successfully

## Related Information

[Capability \[page 225\]](#)

## 4.1.4.5 Delete a Capability

With this method, you delete a `Capability` element with the specified ID.

This method deletes the `Capability` object physically. After the successful deletion, it is not possible to restore the deleted object. Also, along with the `Capability` itself, all actions that have been assigned to the object via the `actions` field are also deleted from the database. That is, they cannot be reused by other `Capability` objects.

If the element is successfully deleted, the server sends HTTP response code 204. Any other code indicates an error.

## Request

URI: `/Capabilities('<ID>')`

HTTP Method: `DELETE`

Permissions: `<auth>.d`

### Request Header Fields

Header	Required	Values
<code>If-Match</code>	Yes	Check if cached version matches server version of the capability

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
ID	Yes	string	Unique identifier of the Capability object	Path

## Request Example

```
/Capabilities('be531d4cfd524f06b8df7c999377fac8')
```

## Response

### Response Status and Error Codes

Code	Description
204	Capability deleted successfully

## Related Information

[Capability \[page 225\]](#)

## 4.1.5 Filter Terms

A method to describe complex object selection criteria

### Overview

A filter term is a formal expression used to describe a set of objects (typically modeled as a Thing) for which a user should have instance authorization. This set of objects can also consist of only one object. In other words, you use a filter term to restrict a user's access rights down to the finest possible granularity. You can go to the extreme of a single user being granted access to one single object, like a car, a coffee machine, or a wind turbine.

The general approach for using filter terms is to describe a group of objects (or an individual object) by the values of its fields. If the field values of an object match the criteria defined in the filter term, a user is granted access to that object to the extent that is defined in the `Capability` object containing the filter term.

#### ❖ Example

You want to set up a capability for users in user group "Control Room Staff" to monitor the technical health of a group of forming presses in a car factory. To select only those instances of industrial machinery that fall



into this category, you define a filter term condition for the Capability that limits the result set to objects that are located in the desired factory, and that have a matching press manufacturer and press type identifier in their property set.

The term is built using Boolean logic and a certain structure. It has to be provided in a structured, JSON-compatible form that can be processed by the filter term framework. Currently, the filter term concept explained in this section is used exclusively for evaluating the `FilterTerm` field in the payload of the `Capability` service.

### Note

Do not confuse the filter term concept described here with the filter that can be used as a query string parameter for almost all of the "Read all..." service methods offered by the SAP Leonardo IoT platform. Although both concepts are similar in that they narrow down an otherwise vast number of objects to a number that can be handled, they are different in purpose as well as in their technical implementation. For more information on the filter query string parameter, see [Filter Conditions \(\\$filter\) \[page 1214\]](#).

## Filter Term Constituents

Filter terms consist of a set of different parts, which constitute the selection criteria defined by the term. The following types of elements exist:

- Filter Term Element
- Filter Term Hierarchy Assignment
- Filter Term Expression
- Filter Term Negation

### Filter Term Element

A filter term of type `FilterTermElement` has the following JSON format:

```
{
  "type" : "FilterTermElement",
  "path" : <Any Path>
  "operator" : <Operator>,
  "value" : <Value>
}
```

Here, `path` describes the data source to be queried in the filter term. It is a structured element that consists of two fields referring to the underlying database entities:

```
{
  "field" : <Field Name>,
  "dimension" : <Table Name>
}
```

To retrieve the possible values for these two fields, use the [Read all CapabilityTypes \[page 224\]](#) service method.

The following operators are supported:

- `EQUAL`, `NOT_EQUAL` (numeric and alphanumeric values; string comparisons are case-sensitive)
- `LOWER` (only numeric values)

- LOWER\_EQUAL (only numeric values)
- GREATER (only numeric values)
- GREATER\_EQUAL (only numeric values)
- STARTS\_WITH (only string values)
- ENDS\_WITH (only string values)
- CONTAINS (only string values)
- MATCHES\_PATTERN (only string values)
- REGEXP (only string values)

### i Note

The MATCHES\_PATTERN operator can be used to search for strings with a search criterion that is defined as a partial equivalent of the searched string, enriched with the following wildcards:

- \* matches any number of characters
- + matches exactly one character

If you want to search for a string that contains one of the wildcard characters, you need to escape the wildcard characters by putting the escape character # in front of it, that is, #\* and #+, respectively. The same applies for searched strings that contain the escape character itself (that is, enter ## in the search string).

For more information on string comparison operations, see [Comparisons Between Character Strings and Byte Strings](#)

The REGEXP operator can be used to define a filter criterion via a regular expression. This operator is only supported for Thing objects, but **not** for entities referred to by other capability types, such as Organization or Person.

For more information on the REGEXP operator, see [LIKE\\_REGEXPR Function \(String\)](#). Note, however, that in the filter term implementation of the REGEXP operator, none of the HANA flag options is used during evaluation of the search expression.

## Filter Term Hierarchy Assignment

A filter term of type FilterTermHierarchyAssignment has the following JSON format:

```
{
  "type" : "FilterTermHierarchyAssignment",
  "path" : <Hierarchy Assignment Path>
  "objectGroupCollections" : [ <ObjectGroupCollection ID>, ... ]
}
```

Here, the value placeholders have the following meaning:

- <Hierarchy Assignment Path>: Path object from the Vocabulary field of the capability type that points to ValueTypeHierarchyAssignment
- <ObjectGroupCollection ID>: ID of an ObjectGroupCollection header object. Here, you can specify an array of one or more object group collection IDs. If a particular object that a user wants to access is assigned to an object group of one or more of these object group collections, the filter term is evaluated to TRUE. That is, the user is granted the required permission and can access the object.

## Filter Term Expression

A filter term of type `FilterTermExpression` has the following JSON format:

```
{
  "type" : "FilterTermExpression",
  "booleanOperator" : <Boolean Operator>,
  "filterTerms" : [ <Any JSON filter term>, ...]
}
```

Here, the `<Boolean Operator>` can have one of the following two values:

- AND
- OR

### Note

When parsing a filter term, the system treats both the AND as well as the OR operator with equal priority. That is, there is no implicit logical precedence of AND over OR as you may know it from many programming languages. As a consequence, you do **not** need to provide extra brackets to influence the internal evaluation logic.

The placeholder `<Any JSON filter term>` refers to the JSON structure of any other filter term. Note that the `filterTerms` part is declared as an array of comma-separated filter term elements.

## Filter Term Negation

A filter term of type `FilterTermNegation` has the following JSON format:

```
{
  "type" : "FilterTermNegation",
  "filterTerm" : <Any JSON filter term>
}
```

Since the `filterTerm` can be a filter term expression, you can logically invert any filter term with one filter term negation, regardless of size and complexity of the expression.

## Example

The following filter term represents a certain Boolean expression given in a form you may be familiar with from the [Where](#) clause used in SQL database queries:

```
NOT[ ( table1.isDeleted EQ false AND table1.isDefault EQ TRUE ) OR
      ( table2.size EQ 3 AND table2.weight EQ 1 ) OR
      ( table2.size EQ 3 AND table2.status EQ OBSOLETE ) ]
```

For a filter term to be used with the `Capability` service, this SQL-like representation must be converted into the following JSON-compatible form:

```
{
  "type": "FilterTermNegation",
  "filterTerm": {
    "type": "FilterTermExpression",
    "booleanOperator": "OR",
    "filterTerms": [
```

```

{
  "type": "FilterTermExpression",
  "booleanOperator": "AND",
  "filterTerms": [{
    "type": "FilterTermElement",
    "path": {
      "field": "isDeleted",
      "dimension": "table1"
    },
    "operator": "EQUAL",
    "value": "false"
  },
  {
    "type": "FilterTermElement",
    "path": {
      "field": "isDefault",
      "dimension": "table1"
    },
    "operator": "EQUAL",
    "value": "true"
  }
  ],
  {
    "type": "FilterTermExpression",
    "booleanOperator": "AND",
    "filterTerms": [{
      "type": "FilterTermElement",
      "path": {
        "field": "size",
        "dimension": "table2"
      },
      "operator": "EQUAL",
      "value": "3"
    },
    {
      "type": "FilterTermElement",
      "path": {
        "field": "weight",
        "dimension": "table2"
      },
      "operator": "EQUAL",
      "value": "1"
    }
  ]
},
{
  "type": "FilterTermExpression",
  "booleanOperator": "AND",
  "filterTerms": [{
    "type": "FilterTermElement",
    "path": {
      "field": "size",
      "dimension": "table2"
    },
    "operator": "EQUAL",
    "value": "3"
  },
  {
    "type": "FilterTermElement",
    "path": {
      "field": "status",
      "dimension": "table2"
    },
    "operator": "EQUAL",
    "value": "Obsolete"
  }
  ]
}
]
}

```

```
}
```

For another example, see the *Request Example* section in [Create a Capability \[page 227\]](#).

## Related Information

[Capability \[page 225\]](#)

[CapabilityType \[page 221\]](#)

[ObjectGroupCollection \[page 241\]](#)

## 4.1.6 ObjectGroupCollection

An `ObjectGroupCollection` represents a subset of object groups in the object group hierarchy

The `ObjectGroupCollection` represents the description of the object group hierarchy. It consists of a header and several item entries. This section is dealing with the header object. For more information on the items of an `ObjectGroupCollection`, see [ObjectGroupCollectionItem \[page 249\]](#).

### → Recommendation

For performance reasons, we recommend limiting the number of object groups that are assigned to an `ObjectGroupCollection` element to no more than 3000. Note that this limit refers to the assigned object groups and **not** to the assigned `ObjectGroupCollectionItem` elements.

Currently, we cannot offer any technical support to ensure that this limit is not exceeded. It is therefore in your own responsibility to keep track of the number of already assigned object groups.

### Base URI:

- Formal description: `http://<server address>[:<port number>][<path>]/ObjectGroupCollections`
- Example for a base URI in a cloud foundry environment: `http://authorization.cfapps.neo.ondemand.com/ObjectGroupCollections`

### Permissions:

`<auth>.c`

`<auth>.r`

`<auth>.d`

## Methods

HTTP Method	Action	URI	Scopes
POST	<a href="#">Create an ObjectGroupCollection</a> [page 243]	/ObjectGroupCollections	<auth>.c
GET	<a href="#">Read an ObjectGroupCollection</a> [page 244]	/ObjectGroupCollections('ID')	<auth>.r
GET	<a href="#">Read all ObjectGroupCollections</a> [page 246]	/ObjectGroupCollections	<auth>.r
DELETE	<a href="#">Delete an ObjectGroupCollection</a> [page 248]	/ObjectGroupCollections('ID')	<auth>.d

### i Note

Although a service method is available to explicitly create an `ObjectGroupCollection` header element, we do not recommend using this method. This is because a header element alone cannot carry any business logic or semantics. Instead, logic and semantics are defined by the items that are associated with the header element. Therefore, creating an `ObjectGroupCollection` header element can also be done implicitly by creating an `ObjectGroupCollectionItem` object with a reference to an `ObjectGroupCollection` header element that does not yet exist in the system.

## Payload

```
{
  "objectGroupCollectionID" : { "type": "string", "maxLength": 32},
  "objectGroupCollectionName" : { "type": "string", "maxLength": 255,
  "required": "true" },
  "tenant" : { "type": "string", "maxLength": 36 },
  "etag" : { "type": "string" }
}
```

### Field Explanation

Name	Description
objectGroupCollectionID	Unique ID of the ObjectGroupCollection.  <div><b>i Note</b> This field is not flagged as mandatory. However, the system ensures that the field always contains a proper value.</div>
objectGroupCollectionName	Descriptive name of the ObjectGroupCollection (for example, "High Pressure Water Pump")

Name	Description
tenant	Name of the tenant to which an ObjectGroupCollection is assigned
etag	Entity tag used to identify a particular instance of an ObjectGroupCollection

## Related Information

[ObjectGroupCollectionItem](#) [page 249]

### 4.1.6.1 Create an ObjectGroupCollection

With this method, you create a new `ObjectGroupCollection` header object

The most convenient way of creating an `ObjectGroupCollection` header object is to create an `ObjectGroupCollectionItem` object with a reference to a header object that does not yet exist. However, for certain use cases, this implicit object creation may not be the preferred approach. Therefore, a dedicated `Create` method is also available although a single header object (i.e., without assigned items) cannot carry any business logic or semantics.

## Request

**URI:** `/ObjectGroupCollections`

**HTTP Method:** `POST`

**Permissions:** `<auth>.r`

## Request Parameters

None.

## Request Example

```
/ObjectGroupCollections
```

## Payload

```
{
```

```

    "objectGroupCollectionID" : { "type": "string", "maxLength": 32},
    "objectGroupCollectionName" : { "type": "string", "maxLength": 255,
    "required": "true" },
    "tenant" : { "type": "string", "maxLength": 36 },
    "etag" : { "type": "string" }
  }

```

## Response

### Response Headers

Header	Description
Location	Path to the newly created ObjectGroupCollection header object

### Response Status and Error Codes

Code	Description
201	ObjectGroupCollection created successfully
403	Forbidden (user is not authorized to create an ObjectGroupCollection)

## Related Information

[ObjectGroupCollection \[page 241\]](#)

### 4.1.6.2 Read an ObjectGroupCollection

Retrieves the ObjectGroupCollection header with the specified ID from the database

With this method, you send a request to the server to retrieve a particular ObjectGroupCollection header object specified by its ID.

#### i Note

With a user assigned to a tenant on T1 level (Cloud Platform Consumer), you can only access ObjectGroupCollection headers that belong to the tenant of the currently logged on user.

## Request

URI: /ObjectGroupCollections ('ID')

HTTP Method: [GET](#)



Permissions: <auth>.r

## Query String Parameters

Parameter	Required	Data Type	Description	Parameter Type
ID	Yes	string	Unique identifier of the ObjectGroup-Collection header object	Path

## Request Example

```
/ObjectGroupCollections('8c0b596b1bd340d3ac7b0072ffb5c8bd')
```

## Response

### Response Status and Error Codes

Code	Description
200	ObjectGroupCollection header retrieved successfully

## Payload

Only media type JSON is supported.

```
{
  "objectGroupCollectionID" : { "type": "string", "maxLength": 32},
  "objectGroupCollectionName" : { "type": "string", "maxLength": 255,
  "required": "true" },
  "tenant" : { "type": "string", "maxLength": 36 },
  "etag" : { "type": "string" }
}
```

## Related Information

[ObjectGroupCollection \[page 241\]](#)

## 4.1.6.3 Read all ObjectGroupCollections

Retrieves a list of `ObjectGroupCollection` header objects from the database

With this method, you send a request to the server to retrieve a subset of all `ObjectGroupCollection` header objects, according to the filter criteria provided. For a collective request for a set of `ObjectGroupCollection` header objects, you have these options:

- You can sort and restrict the set of `ObjectGroupCollection` headers to be retrieved based on the following fields:
  - `tenant`
  - `objectGroupCollectionID`
  - `objectGroupCollectionName`
- You can define that the set of matching `ObjectGroupCollection` headers shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

If the list of `ObjectGroupCollection` headers is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

### i Note

Retrieving high numbers of objects from the database with only one collective GET request can lead to high system load and a significant increase of response times. Therefore, we highly recommend limiting the maximum number of returned objects to no more than 500 per method call. This can easily be accomplished with the help of the `$top` and `$skip` query parameters.

To avoid unwanted performance decline, collective GET requests are only supported up to a maximum of 500 objects that can be retrieved with one single call. Therefore, in use cases where it is crucial for you to ensure that all existing objects have been retrieved, you need to verify this by setting up a loop construct in your source code to repeat the request until the service method returns no further objects.

## Request

**URI:** `/ObjectGroupCollections`

**HTTP Method:** `GET`

**Permissions:** `<auth>.r`

### Query String Parameters

Name	Required	Type	Description
<code>\$skip</code>	No	integer	Number of the first n records to be excluded from the result set
<code>\$top</code>	No	integer	Number of records to include in the result set

Name	Required	Type	Description
\$orderby	No	string	Field name to be used for sorting the result set in ascending or descending order
\$filter	No	string	Filter condition to be applied; valid fields are tenant, objectGroupCollectionID, and objectGroupCollectionName

### Note

For the filter parameter, the following applies:

- Defining a filter is **not** mandatory for this service method.
- The filter behavior is different depending on the tenant level of the currently logged on user:
  - User is assigned to a T0 tenant (Cloud Platform Provider): Defining a filter definition can be used to restrict the result set to the ObjectGroupCollection headers belonging to the specified tenant. Otherwise, all ObjectGroupCollection headers are returned.
  - User is assigned to a T1 tenant (Cloud Platform Consumer): Only the ObjectGroupCollection headers belonging to the user's tenant are returned. If a different tenant is specified in the filter, the system returns an empty result list.

## Request Example

```
/ObjectGroupCollections
```

Retrieves all ObjectGroupCollection headers that are available for the currently logged on user.

```
/ObjectGroupCollections?$top=25
```

Retrieves the first 25 ObjectGroupCollection headers. Data is retrieved in page mode to reduce network load and response times.

```
/ObjectGroupCollections?$skip=10&$top=20
```

Retrieves 20 ObjectGroupCollection header objects from the database where the first 10 ObjectGroupCollection header objects are omitted. That is, the retrieved headers are those with sequence numbers 11 through 30.

```
/ObjectGroupCollections?$filter=tenant eq 'Customer-test001'&
$orderby=objectGroupCollectionName
```

Retrieves all ObjectGroupCollection headers from the database that belong to the specified tenant. The list of headers is alphabetically sorted based on their names.

## Response

Format: *JSON*

## Payload

```
{
  "objectGroupCollectionID" : { "type": "string", "maxLength": 32},
  "objectGroupCollectionName" : { "type": "string", "maxLength": 255,
"required": "true" },
  "tenant" : { "type": "string", "maxLength": 36 },
  "etag" : { "type": "string" }
}
```

## Response Status and Error Codes

Code	Reason
200	ObjectGroupCollection headers retrieved successfully

## Related Information

[ObjectGroupCollection \[page 241\]](#)

### 4.1.6.4 Delete an ObjectGroupCollection

With this method, you delete an ObjectGroupCollection header with the specified ID.

This method deletes the ObjectGroupCollection header physically. After the successful deletion, it is not possible to restore the deleted object. All ObjectGroupCollectionItem elements that are referring to the deleted header are automatically deleted by the system to preserve data integrity (cascaded deletion).

If the ObjectGroupCollection header is successfully deleted, the server sends HTTP response code 204. Any other code indicates an error.

## Request

URI: /ObjectGroupCollections ('ID')

HTTP Method: [DELETE](#)

Permissions: <auth>.d

## Request Header Fields

Header	Required	Values
If-match	Yes	Check if cached version matches server version of the ObjectGroupCollection header

## Query String Parameters

Parameter	Required	Data Type	Description	Parameter Type
ID	Yes	string	unique identifier of the ObjectGroupCollection header	Path

## Request Example

```
/ObjectGroupCollections('8c0b596b1bd340d3ac7b0072ffb5c8bd')
```

Deletes the ObjectGroupCollection header with the specified ID.

## Response

### Response Status and Error Codes

Code	Description
204	ObjectGroupCollection header deleted successfully

## Related Information

[ObjectGroupCollection \[page 241\]](#)

[ObjectGroupCollectionItem \[page 249\]](#)

## 4.1.7 ObjectGroupCollectionItem

An `ObjectGroupCollectionItem` is a pointer to an object group in an `ObjectGroupCollection`.

The `ObjectGroupCollectionItem` serves as a connection between an `ObjectGroup` and an `ObjectGroupCollection` header element. One `ObjectGroupCollection` header element can have many `ObjectGroupCollectionItem` elements assigned, while each `ObjectGroupCollectionItem` element is assigned to exactly one `ObjectGroup`. For more information on the header element of an `ObjectGroupCollection`, see [ObjectGroupCollection \[page 241\]](#).

This architecture, at first glance, might bring up the question why the `ObjectGroupCollectionItem` is needed at all as an intermediate element between the `ObjectGroupCollection` header and the `ObjectGroup`. The answer is that the `ObjectGroupCollectionItem` is needed as an element that controls whether the sub-ordinate object groups that are descending from the referenced object group are also affected by the assigned capability or not. This is controlled with the help of the `isSubtreeSelected` flag, which can be either true or false.

Regarding cross-tenant maintenance, the following applies:

- Users that are assigned to a tenant of a cloud platform consumer (T1) can only access items that belong to the tenant where they are logged on to.

- Users that are assigned to a tenant of a cloud platform provider (T0) **and** who have been granted cross-tenant maintenance authorization can access items from all tenants, regardless of the tenant where they are logged on to.

#### Base URI:

- Formal description: `http://<server address>[:<port number>][<path>]/ObjectGroupCollectionItems`
- Example for a base URI in a cloud foundry environment: `http://authorization.cfapps.neo.ondemand.com/ObjectGroupCollectionItems`

#### Permissions:

<auth>.c

<auth>.r

<auth>.u

<auth>.d

## Methods

HTTP Method	Action	URI	Scopes
POST	<a href="#">Create an ObjectGroupCollectionItem [page 251]</a>	/ObjectGroupCollectionItems	<auth>.c
GET	<a href="#">Read an ObjectGroupCollectionItem [page 253]</a>	/ObjectGroupCollectionItems (ogc='ogcID', og='ogID')	<auth>.r
GET	<a href="#">Read all ObjectGroupCollectionItems [page 254]</a>	/ObjectGroupCollectionItems	<auth>.r
PUT	<a href="#">Update an ObjectGroupCollectionItem [page 256]</a>	/ObjectGroupCollectionItems (ogc='ogcID', og='ogID')	<auth>.u
DELETE	<a href="#">Delete an ObjectGroupCollectionItem [page 258]</a>	/ObjectGroupCollectionItems (ogc='ogcID', og='ogID')	<auth>.d

## Payload

Only media type JSON is supported.

```
{
  "objectGroupCollectionID" : { "type": "string", "maxLength": 32 },
  "objectGroupCollectionName" : { "type": "string", "maxLength": 255,
    "required": "true" },
  "objectGroupID" : { "type": "string", "maxLength": 32,
    "required": "true" },
}
```

```

    "isSubtreeSelected" : { "type": "boolean" },
    "etag"              : { "type": "string" }
  }

```

Field Explanation

Name	Description
objectGroupCollectionID	ID of the ObjectGroupCollection header element to which the item is assigned
objectGroupID	ID of the ObjectGroup element to which the item refers
isSubtreeSelected	Indicates whether the referenced object group and its descendants in the object group hierarchy are affected by the assigned capability or not. If this flag is set to <code>false</code> this setting is propagated to all descendants as long as no lower level object group is found with deviating capability settings.
etag	Identifier of the ObjectGroupCollectionItem instance version

## Related Information

[ObjectGroupCollection](#) [page 241]

[ObjectGroup](#) [page 206]

### 4.1.7.1 Create an ObjectGroupCollectionItem

With this method, you create a new item of an `ObjectGroupCollection`.

You use an `ObjectGroupCollectionItem` element to associate an `ObjectGroup` to the superordinate `ObjectGroupCollection` header element. In other words, the `ObjectGroupCollectionItem` serves as a link between the `ObjectGroupCollection` header and an `ObjectGroup` element that you want to refer to via the `ObjectGroupCollectionItem`.

The following mechanisms of this method are worth knowing:

- If the `ObjectGroupCollection` header element that is passed to the method already exists, the new `ObjectGroupCollectionItem` is added to the list of items associated with that header.
- If the `ObjectGroupCollection` header element that is passed to the method does **not** already exist, it is automatically created along with the new item, and the two new elements are associated with each other.
- During creation of an `ObjectGroupCollectionItem`, the `ObjectGroupCollection` header element is locked and cannot be changed by other processes.

## Request

**URI:** `/ObjectGroupCollectionItems`

HTTP Method: *POST*

Permissions: <auth>.c

## Request Parameters

None.

## Request Example

```
/ObjectGroupCollectionItems
```

## Payload

```
{
  "objectGroupCollectionID" : { "type": "string", "maxLength": 32 },
  "objectGroupCollectionName" : { "type": "string", "maxLength": 255,
    "required": "true" },
  "objectGroupID" : { "type": "string", "maxLength": 32,
    "required": "true" },
  "isSubtreeSelected" : { "type": "boolean" },
  "etag" : { "type": "string" }
}
```

## Response

### Response Headers

Header	Description
Location	Path to the newly created ObjectGroupCollectionItem

### Response Status and Error Codes

Code	Description
201	ObjectGroupCollectionItem created successfully
403	Forbidden (user is not authorized to create an ObjectGroupCollectionItem)

## Related Information

[ObjectGroupCollectionItem \[page 249\]](#)



## 4.1.7.2 Read an ObjectGroupCollectionItem

Retrieves an `ObjectGroupCollectionItem` with the specified ID from the database

An `ObjectGroupCollectionItem` is identified by the combination of its own ID and the ID of the `ObjectGroup` element to which it refers. Therefore, you need to specify both IDs as request parameters of this method.

### Request

**URI:** /

```
ObjectGroupCollectionItems(objectGroupCollectionID='<ogcID>',objectGroupID='<ogID>')
```

**HTTP Method:** [GET](#)

**Permissions:** <auth>.r

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
objectGroupCollectionID	Yes	string	ID of the ObjectGroupCollection header element to which the item is assigned	Path
objectGroupID	Yes	string	ID of the ObjectGroup element to which the item refers	Path

### Request Example

```
/ObjectGroupCollectionItems(objectGroupCollectionID='8c0b596b1bd340d3ac7b0072ffb5c8bd',objectGroupID='be531d4cfd524f06b8df7c999377fac8')
```

### Response

#### Response Status and Error Codes

Code	Description
200	Item retrieved successfully

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "objectGroupCollectionID" : { "type": "string", "maxLength": 32 },
  "objectGroupCollectionName" : { "type": "string", "maxLength": 255,
  "required": "true" },
  "objectGroupID" : { "type": "string", "maxLength": 32,
  "required": "true" },
  "isSubtreeSelected" : { "type": "boolean" },
  "etag" : { "type": "string" }
}
```

## Related Information

[ObjectGroupCollectionItem \[page 249\]](#)

### 4.1.7.3 Read all ObjectGroupCollectionItems

Retrieves a list of `ObjectGroupCollectionItem` objects from the database

With this method, you send a request to the server to retrieve a subset of all `ObjectGroupCollectionItem` objects, according to the filter criteria provided. For a collective request for a set of `ObjectGroupCollectionItem` objects, you have these options:

- You can sort and restrict the set of `ObjectGroupCollectionItem` objects to be retrieved by filter criteria as shown in the table below.
- You can define that the set of matching `ObjectGroupCollectionItem` objects shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

#### i Note

Retrieving high numbers of objects from the database with only one collective GET request can lead to high system load and a significant increase of response times. Therefore, we highly recommend limiting the maximum number of returned objects to no more than 500 per method call. This can easily be accomplished with the help of the `$top` and `$skip` query parameters.

To avoid unwanted performance decline, collective GET requests are only supported up to a maximum of 500 objects that can be retrieved with one single call. Therefore, in use cases where it is crucial for you to ensure that all existing objects have been retrieved, you need to verify this by setting up a loop construct in your source code to repeat the request until the service method returns no further objects.

## Request

**URI:** /ObjectGroupCollectionItems

**HTTP Method:** [GET](#)

**Permissions:** <auth>.r

### Query String Parameters

Name	Required	Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set
\$orderby	No	string	Field name to be used for sorting the result set in ascending or descending order
\$filter	No	string	Filter condition to be applied; valid fields are objectGroupCollectionHeader/tenant, objectGroupCollectionHeader/name, objectGroupCollectionID, and objectGroupID

#### i Note

For the filter parameter, the following applies:

- Defining a filter is mandatory for this service method.
- The filter behavior is different depending on the tenant level of the currently logged on user:
  - User is assigned to a T0 tenant (Cloud Platform Provider): Defining a filter definition can be used to restrict the result set to the ObjectGroupCollection headers belonging to the specified tenant. Otherwise, all ObjectGroupCollectionItem objects are returned.
  - User is assigned to a T1 tenant (Cloud Platform Consumer): Only the ObjectGroupCollectionItem objects belonging to the user's tenant are returned. If a different tenant is specified in the filter, the system returns an empty result list.

### Request Parameters

None.

### Request Example

```
/ObjectGroupCollectionItems
```

Retrieves all ObjectGroupCollectionItem objects that are available for the currently logged on user. Note that this unrestricted request is not recommended due to high system load.

```
/ObjectGroupCollectionItems?$top=25
```

Retrieves the first 25 ObjectGroupCollectionItem objects from the database.

```
/ObjectGroupCollectionItems?$filter=objectGroupID eq  
'8c0b596b1bd340d3ac7b0072ffb5c8bd'
```

Retrieves all `ObjectGroupCollectionItem` objects that refer to the specified `ObjectGroup`.

```
/ObjectGroupCollectionItems?$skip=10&$top=20
```

Retrieves 20 `ObjectGroupCollectionItem` objects from the database where the first 10 items are omitted. That is, the retrieved items are those with sequence numbers 11 through 30.

## Response

### Payload

```
{
  "objectGroupCollectionID" : { "type": "string", "maxLength": 32 },
  "objectGroupCollectionName" : { "type": "string", "maxLength": 255,
  "required": "true" },
  "objectGroupID" : { "type": "string", "maxLength": 32,
  "required": "true" },
  "isSubtreeSelected" : { "type": "boolean" },
  "etag" : { "type": "string" }
}
```

### Response Status and Error Codes

Code	Description
200	Items retrieved successfully

## Related Information

[ObjectGroupCollectionItem \[page 249\]](#)

### 4.1.7.4 Update an ObjectGroupCollectionItem

With this method, you modify the specified `ObjectGroupCollectionItem`.

With this method, you modify the specified `ObjectGroupCollectionItem`. With `isSubtreeSelected` being the only field of an `ObjectGroupCollectionItem` that can be modified, calling this service method lets you either extend the scope of authorizations from one particular object group to all of its descendants or vice versa.

#### i Note

There is only one field in the service payload that you can modify with this method, namely the boolean value of `isSubtreeSelected`. Any attempt to change other field values will lead to an error.

## Request

URI: /

```
ObjectGroupCollectionItems(objectGroupCollectionID='<ogcID>',objectGroupID='<ogID>')
```

HTTP Method: *PUT*

Permissions: <auth>.u

### Request Header Fields

Header	Required	Description
If-Match	Yes	Check if cached version matches server version of the ObjectGroupCollectionItem

### Query String Parameters

Parameter	Required	Data Type	Description	Parameter Type
ogc	Yes	string	ID of the ObjectGroupCollection header	Path
og	Yes	string	ID of the ObjectGroup to which the item refers	Path

### Request Example

```
/
ObjectGroupCollectionItems(objectGroupCollectionID='8c0b596b1bd340d3ac7b0072ffb5c8bd',objectGroupID='c35cef3368cf439884823369479af8b1')
```

### Payload

Only media type JSON is supported.

```
{
  "objectGroupCollectionID" : { "type": "string", "maxLength": 32 },
  "objectGroupCollectionName" : { "type": "string", "maxLength": 255,
"required": "true" },
  "objectGroupID" : { "type": "string", "maxLength": 32,
"required": "true" },
  "isSubtreeSelected" : { "type": "boolean" },
  "etag" : { "type": "string" }
}
```

#### i Note

As mentioned above, there is only one field in the service payload that you can modify with this method, namely the boolean value of `isSubtreeSelected`. Any attempt to change other field values will lead to an error.

#### Field Explanation

Name	Description
objectGroupCollectionID	ID of the ObjectGroupCollection header element to which the item is assigned
objectGroupID	ID of the ObjectGroup element to which the item refers
isSubtreeSelected	Indicates whether the referenced object group and its descendants in the object group hierarchy are affected by the assigned capability or not. If this flag is set to <code>false</code> this setting is propagated to all descendants as long as no lower level object group is found with deviating capability settings.
etag	Identifier of the ObjectGroupCollectionItem instance version

## Response

### Response Status and Error Codes

Code	Description
200	ObjectGroupCollectionItem updated successfully

## Related Information

[ObjectGroupCollectionItem \[page 249\]](#)

### 4.1.7.5 Delete an ObjectGroupCollectionItem

With this method, you delete an `ObjectGroupCollectionItem` element with the specified ID.

This method deletes the `ObjectGroupCollectionItem` object physically. After the successful deletion, it is not possible to restore the deleted object.

#### Note

In the [ObjectGroupCollection \[page 241\]](#) section, we have explained that a header is created implicitly by creating an `ObjectGroupCollectionItem` element with a reference to an `ObjectGroupCollection` header that doesn't yet exist. This might lead to the assumption that deleting the last `ObjectGroupCollectionItem` associated with a particular `ObjectGroupCollection` header would also implicitly delete that header object. However, triggering such an automated, cascaded deletion of the associated `ObjectGroupCollection` header is **not** implemented. Instead, after deleting the last item the `ObjectGroupCollection` header remains in the system as an empty collection. It can be reused by creating new `ObjectGroupCollectionItem` objects with a reference to that header.

If the element is successfully deleted, the server sends HTTP response code 204. Any other code indicates an error.

## Request

**URI:** /

ObjectGroupCollectionItems(objectGroupCollectionID='<ogcID>', objectGroupID='<ogID>')

**HTTP Method:** [DELETE](#)

**Permissions:** <auth>.d

### Request Header Fields

Header	Required	Description
If-Match	Yes	Check if cached version matches server version of the ObjectGroupCollectionItem

### Query String Parameters

Parameter	Required	Data Type	Description	Parameter Type
objectGroupCollectionID	Yes	string	ID of the ObjectGroupCollection header	Path
objectGroupID	Yes	string	ID of the ObjectGroup to which the item refers	Path

### Request Example

```
/ObjectGroupCollectionItems(objectGroupCollectionID='8c0b596b1bd340d3ac7b0072ffb5c8bd',objectGroupID='c35cef3368cf439884823369479af8b1')
```

## Response

### Response Status and Error Codes

Code	Description
204	ObjectGroupCollectionItem successfully deleted

## Related Information

[ObjectGroupCollectionItem \[page 249\]](#)

### 4.1.8 Capabilities

The following tables describe the capabilities required per HTTP method:

#### Capabilities for Accessing Thing, Event, and File Services

HTTP Methods (APIs) with Business Partner Dependency	Supported Capabilities		
	read	write	delete
POST /Things	n/a	X	n/a
GET /Things	X	n/a	n/a
POST /Things ('<Thing ID>')	n/a	X	n/a
DELETE /Things ('<Thing ID>')	n/a	n/a	X
GET /Things ('<Thing ID>')	X	n/a	n/a
GET /Things ('<Thing ID>')/Configuration	X	n/a	n/a
GET /Things/PropertySet/<propertySetType name>	X	n/a	n/a
PUT /Things ('<Thing ID>')/<thingType>/<propertySet ID>	n/a	X	n/a
GET /Things ('<Thing ID>')/<thingType>/<propertySet ID>	X	n/a	n/a
DELETE /Things ('<Thing ID>')/<thingType>/<propertySet ID>	n/a	n/a	X



HTTP Methods (APIs) with Business Partner Dependency	Supported Capabilities		
	read	write	delete
GET /Composite/Things	X	n/a	n/a
GET /Snapshot	X	n/a	n/a
GET M4 algorithm	X	n/a	n/a
POST /Events	n/a	X	n/a
GET /Events	X	n/a	n/a
POST /Events ('<Event ID>')	n/a	X	n/a
GET /Events ('<Event ID>')	X	n/a	n/a
DELETE /Events ('<Event ID>')	n/a	n/a	X
POST /Events.SetStatus(newStatus=<status>)	n/a	X	n/a
PUT /Files/Things/<Thing ID>/<path>/<filename>	n/a	X	n/a
GET /Files/Things/<Thing ID>/<path>/<filename>	X	n/a	n/a
DELETE /Files/Things/<Thing ID>/<path>/<filename>	n/a	n/a	X

## Capabilities for Accessing Business Partner Services

### Basic Services

For the basic services that belong to the business partner domain (that is, `Person` and `Organization`), the following simple rules apply:

- `POST`: Write capability
- `GET` (both single and collective request): Read capability
- `PUT`: Write capability
- `DELETE`: Delete capability

## Services Used for Establishing Relationships Between Business Partners

A number of services in the SAP Leonardo IoT platform are used for establishing a connection between different business partners (for example, a person is employee of an organization) or between a person and the user object that is used to authenticate that person against the system. These services are:

- [BPRelationship \[page 87\]](#)
- [BPUserRelationship \[page 116\]](#)

Surprisingly, write access to these services actually does **not** mean changing the respective relationship object. This is because the relationship services are **not** reflected by a corresponding entity in the database. Rather, the relationship between two objects is stored as mutual references of the two objects involved. Therefore, the necessary capabilities for relationship services are as follows:

HTTP Methods (APIs) for Relationships	Supported Capabilities		
	read	write	delete
POST /BPRelationships	n/a	n/a	n/a
PUT [ /Persons('<ID1>')   /Organizations('<ID1>') ]		X	
PUT [ /Persons('<ID2>')   /Organizations('<ID2>') ]		X	
GET /BPRelationships('relationshipID')	n/a	n/a	n/a
GET [ /Persons('<ID1>')   /Organizations('<ID1>') ]	X		
GET [ /Persons('<ID2>')   /Organizations('<ID2>') ]	X		
DELETE /BPRelationships('relationshipID')	n/a	n/a	n/a
PUT [ /Persons('<ID1>')   /Organizations('<ID1>') ]		X	
PUT [ /Persons('<ID2>')   /Organizations('<ID2>') ]		X	
POST /BPUserRelationships	n/a	n/a	n/a
PUT [ /Persons('<ID>')   /Organizations('<ID>') ]		X	
GET /BPUserRelationships('relationshipID')	n/a	n/a	n/a
GET [ /Persons('<ID>')   /Organizations('<ID>') ]	X		
DELETE /BPUserRelationships('relationshipID')	n/a	n/a	n/a
PUT [ /Persons('<ID>')   /Organizations('<ID>') ]		X	

## 4.2 Functional Authorization

Authorization in the application router and container are handled by scopes. The functional authorizations are organized in scopes.

### Scopes

- Scopes cover business users' authorizations in a specific Java application. They are deployed, for example, when you deploy an update of the application. The security descriptor file `xs-security.json` contains the application-specific "local" scopes (and, if applicable, also "foreign" scopes, which are valid for other defined applications).
- Scopes are assigned to users by means of security roles, which are mapped to the user group(s) to which the user belongs. Scopes are used for authorization checks by the application router.
- To assign scopes to an application, you need to perform the following steps
  - Create an instance of the User Account and Authentication (UAA) service. You can use the service broker to create an instance of the UAA service. This ensures the creation of a new OAuth 2.0 client in the UAA
  - Assign scopes for the application. Scopes are configured and assigned in the application descriptor file `xs-security.json`

### Application Container

The container security API includes the `hasScope` method that allows you to programmatically check whether the OAuth 2.0 access token used for the current request has the appropriate scope. Among other things, the access token contains a list of scopes that the user is allowed to access in the current context. Each scope name must be unique in the context of the current UAA installation.

### Scopes and Application Names

In this API reference documentation, the scopes that are required for executing a service are listed for each service method in the *Permissions* part of the method documentation. Since the scopes are defined by the various applications that make up the SAP Leonardo IoT, each scope name consists of the application and the activity that is controlled by the scope. A formal description of a fully qualified scope looks as follows:

```
<xs_appname>[.<additional_resource>].<action>
```

Here is an explanation of the different parts of this formal description:

- `<xs_appname>`: This name is defined during deployment. Within a productive system landscape, this name does not change over time, but it may vary from landscape to landscape. For example, the name used in a development landscape may not be the same as in the test landscape, and both may differ from the name in the productive landscape.
- `<action>`: Self-explanatory. According to the CRUD pattern known from databases, the following actions are supported: Create, Read, Update, Delete.

#### i Note

The four supported activities are abbreviated to their initial letter in lowercase (that is, `c`, `r`, `u`, and `d`). Some complex operations in the Leonardo IoT platform require that a user performing the operation

needs a high number of permissions, which have to be passed to the system all at once. The short abbreviations make sure that this does not result in a field length overflow.

- `<additional_resource>`: Identifier of a particular resource within an application for which a scope is defined. For example, the `<thing>.event.r` scope is required for a user to read the events that have been logged for a thing, where `event` is a resource that belongs to the `thing` application.

## i Note

In this API reference documentation, we present the application-specific part of the scopes in a simplified form that makes it easy to understand the application in question, but still needs to be adapted to the particular `<xs_appname>` that is in effect in your system landscape. The following symbolic names are used for this:

Symbolic Application Names for Scopes

Application	Symbolic Name
authorization	<code>&lt;auth&gt;</code>
business-partner	<code>&lt;bp&gt;</code>
location	<code>&lt;loc&gt;</code>
tenant-administration	<code>&lt;tenant&gt;</code>
thing	<code>&lt;thing&gt;</code>

Keep in mind that the angle brackets are used here to indicate the symbolic character of the name of the application defining the scope. In a real usage scenario, the entire symbolic name (including the angle brackets) must be replaced by the technical application name that has been chosen for your system landscape.

## Application Names and Service URIs

In the previous section, we have explained the relation between application names and scope qualifiers. A similar mechanism is in place (although not relevant for authorization) with respect to the service endpoints that are given in the *Base URI* part of each service overview chapter. Here, the formal description of a fully qualified URI looks like this:

```
http://<application name>.<application domain>[:<port number>][/<path>]
```

Like in the *Scopes and Application Names* section above, the `<application name>` as well as the `<application domain>` part are subject to arbitrary decisions of system administrators and cannot be considered a stable and uniform URI part across all installations around the globe. The reliable part, however, is the fact that by default, the `<application name>` part is identical with the application names listed above.

## i Note

In certain scenarios, the service URIs become longer as there could be too many request parameters in the URL to fetch the desired result set in the response payload. The long service URI results in large header size and the execution of the service fails with an error status code 400. Header size is limited to 16k. To overcome the long service URI, you can break the request into multiple requests.

For example, to [Read Reference Properties Data for a List of Property Sets \[page 864\]](#), you use the request URL `https://appiot-mds.cfapps.eu10.hana.ondemand.com/Things('503B8711B5D0445688E7FF7ACFC2177A')/core.automobiles:car/ReferenceProperties/'wheel_rr,Pressure&wheel_rr,RotationSpeed&wheel_fl,Pressure&wheel_fl,RotationSpeed'?timerange2015-03-10T08:00:00.000Z-2018-08-09T12:00:00.000Z`. The request might fail when there are huge number of reference properties in the request URL. To avoid larger header size, you can split a single request and hence, reduce the number of properties per request.

Here are some examples of service URIs to access the services:

#### Service URIs

Application	Service URI
<a href="#">Advanced Thing List [page 1136]</a> : OData	<code>https://advancedlist-thing-sap.cfapps.eu10.hana.ondemand.com/CompositeThings</code>
Authorization: <a href="#">ObjectGroup [page 206]</a>	<code>https://authorization.cfapps.eu10.hana.ondemand.com/ObjectGroups</code>
Business Partner: <a href="#">Organization [page 28]</a>	<code>https://business-partner.cfapps.eu10.hana.ondemand.com/Organizations</code>
Business Partner: <a href="#">Person [page 55]</a>	<code>https://business-partner.cfapps.eu10.hana.ondemand.com/Persons</code>
<a href="#">Composite Events: OData Service [page 1062]</a> : OData	<code>https://composite-events-odata.cfapps.eu10.hana.ondemand.com/CompositeEvents</code>
<a href="#">Composite Things: REST Service [page 1043]</a> : REST	<code>https://composite-things.cfapps.eu10.hana.ondemand.com/Composite/Things/PropertySet</code>
<a href="#">Composite Things: OData Service [page 1047]</a> : OData	<code>https://composite-things-odata.cfapps.eu10.hana.ondemand.com/CompositeThings</code>
<a href="#">Composite Thing Configuration for a Specific Thing Hierarchy: OData Service [page 1093]</a> : OData	<code>https://cs-hierarchy-meta-sap.cfapps.eu10.hana.ondemand.com/CompositesThingHierarchy</code>
<a href="#">Event Configuration [page 512]</a> : OData	<code>https://events-sap.cfapps.eu10.hana.ondemand.com/ES</code>
<a href="#">Package Configuration [page 337]</a> : OData	<code>https://config-package-sap.cfapps.eu10.hana.ondemand.com/Package</code>
<a href="#">Read Thing Configuration Details for a Thing [page 734]</a>	<code>https://appiot-mds.cfapps.eu10.hana.ondemand.com/Configuration</code>

Application	Service URI
<a href="#">Event List: OData Service [page 1172]</a>	https://bpanalytics-event-sap.cfapps.eu10.hana.ondemand.com/BPAnalytics/EventType/v1/<event type name>
<a href="#">Standard or Predefined Events [page 983]: REST</a>	https://apiot-mds.cfapps.eu10.hana.ondemand.com/Events
<a href="#">Events Aggregate: OData Service [page 1181]</a>	https://events-aggregate-sap.cfapps.eu10.hana.ondemand.com/EventsAggregate/v1/Events?timeRange=<from time>-<to time>&divisions=<divisions>
<a href="#">File [page 1033]: REST</a>	https://apiot-fs.cfapps.eu10.hana.ondemand.com/Files
<a href="#">Credentials [page 753]</a>	https://tm-data-mapping.cfapps.eu10.hana.ondemand.com/v1/credentials
<a href="#">Mapping [page 761]</a>	https://tm-data-mapping.cfapps.eu10.hana.ondemand.com/v1/mappings
<a href="#">Assignment [page 785]</a>	https://tm-data-mapping.cfapps.eu10.hana.ondemand.com/v1/assignments
<a href="#">Custom Master Data [page 598]</a>	https://config-customdata-sap.cfapps.eu10.hana.ondemand.com/CustomDataConfiguration
<a href="#">Custom Master Data Composite Services [page 667]</a>	https://composite-custom-master-data-sap.cfapps.eu10.hana.ondemand.com/CompositeMasterData/v1/Thingshttps://composite-custom-master-data-sap.cfapps.eu10.hana.ondemand.com/CompositeMasterData/Details/v1/Things
<a href="#">Actions [page 688]</a>	https://sap-iot-noah-live-action-modeler.cfapps.eu10.hana.ondemand.com/envPing
<a href="#">Thing Detail [page 1121]: OData</a>	https://details-thing-sap.cfapps.eu10.hana.ondemand.com/CompositeThings/v1/<thing type name>
<a href="#">Thing Hierarchy [page 945]: OData</a>	https://apiot-thing-hierarchy.cfapps.eu10.hana.ondemand.com/ThingHierarchy
<a href="#">Thing Configuration [page 352]: OData</a>	https://config-thing-sap.cfapps.eu10.hana.ondemand.com/ThingConfiguration

Application	Service URI
<a href="#">Onboarding Things [page 715]</a> : REST	https://appiot-mds.cfapps.eu10.hana.ondemand.com/Things
<a href="#">Thing Time Series Data: OData Service [page 917]</a>	https://analytics-thing-sap.cfapps.eu10.hana.ondemand.com/<property set type name>
<a href="#">Time Series Aggregate Store [page 890]</a> : REST	https://appiot-mds.cfapps.eu10.hana.ondemand.com/Things
<a href="#">Time Series Cold Store [page 910]</a> : REST	https://appiot-coldstore.cfapps.eu10.hana.ondemand.com/Things
<a href="#">Time Series Store [page 872]</a> : REST	https://appiot-mds.cfapps.eu10.hana.ondemand.com/Things
<a href="#">Value Helps [page 1238]</a>	https://business-partner.cfapps.eu10.hana.ondemand.com/<Value Help>

## Attributes

You can define attributes to perform checks based on a source that is not yet defined. For example, in xs-security.json file, the check is based on a cost center, whose name is not known because it differs according to context.

## Role Templates

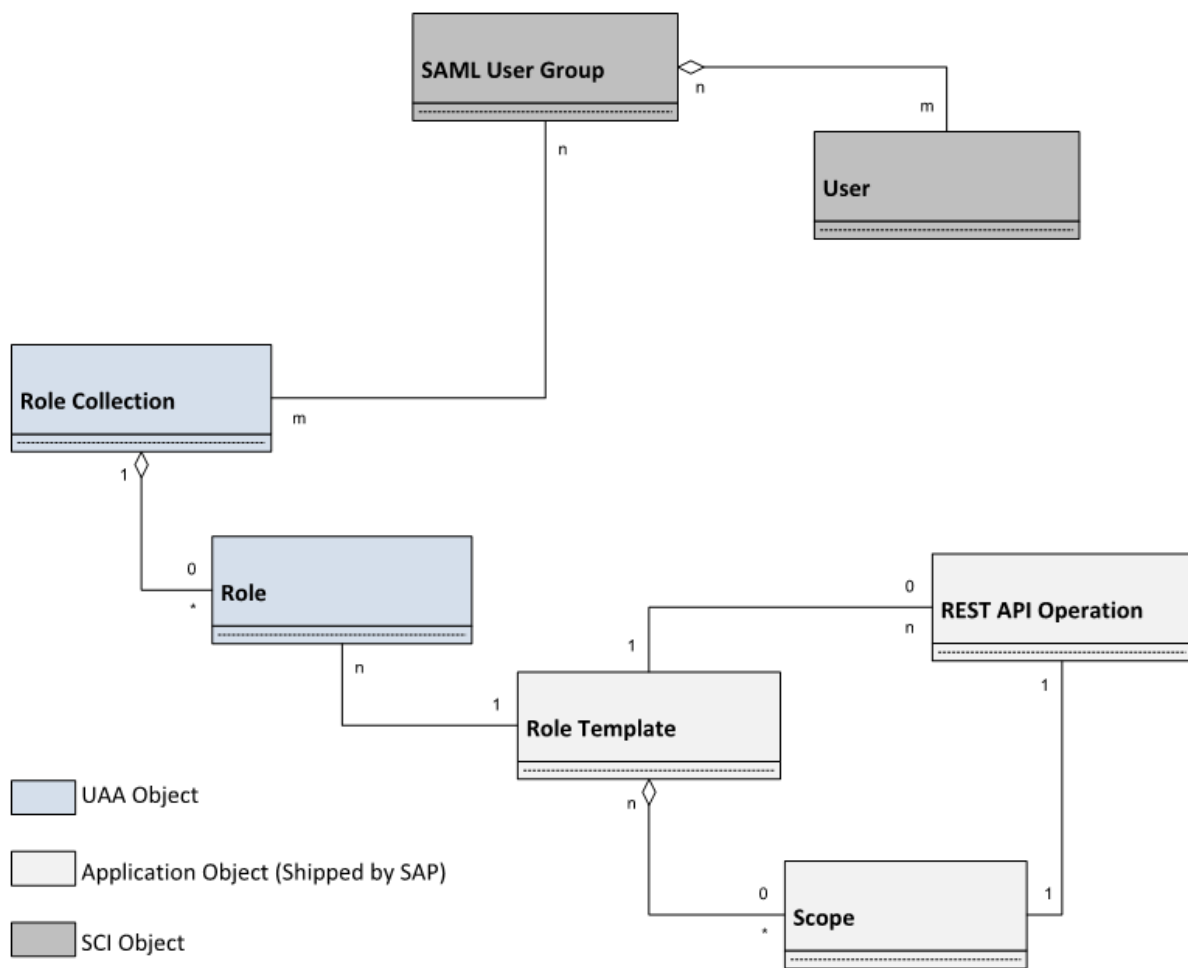
A role template describes a role and any attributes that apply to the role.

Scopes provide a set of role templates for a named application. The role templates contain authorizations for business users' activities such as viewing, editing, or deleting data. Information retrieved from the user's identity (such as department or cost center) is stored in attributes. After you have created the role templates and deployed them to the relevant application, it is an administrator's task to do the following:

- Use the role templates to build roles.
- Aggregate roles into role collections.
- Assign the role collections to business users in the application.

## Entities and Relationships

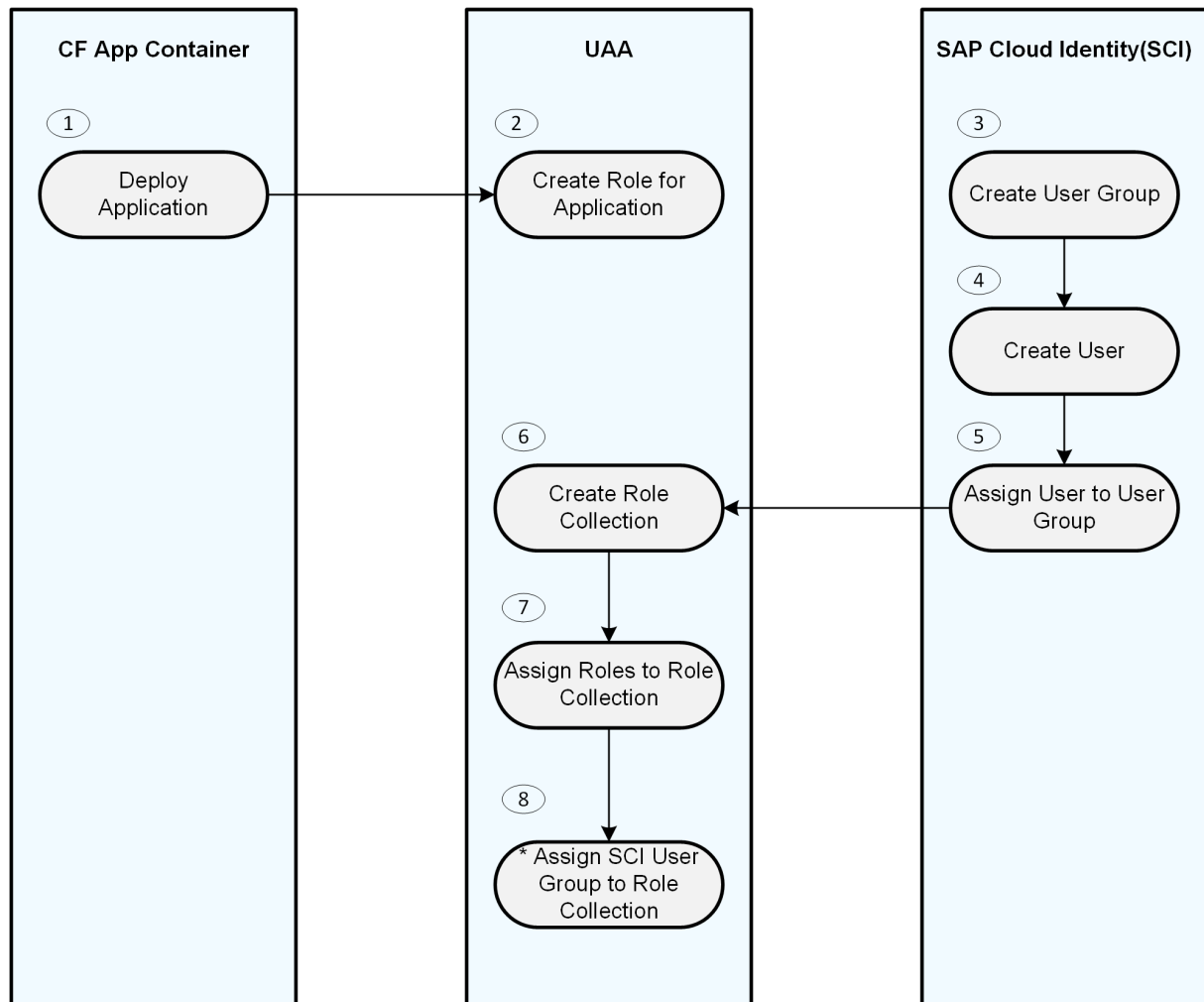
The following diagram represents the different entities and the relationships between each one.



## Setting Up Security Artifacts

Developers create authorization information for business users in their environment; the information is deployed in an application and made available to administrators who complete the authorization setup and assign the authorizations to business users. Developers store authorization information as design-time role templates in the security descriptor file `xs-security.json`. Using the UAA service broker, they deploy the security information. The administrators view the authorization information in role templates, which they use as part of the run-time configuration. The administrators use the role templates to build roles, which are aggregated in role collections. The role collections are assigned, in turn, to business users. After the deployment of the authorization artifacts as role templates, the administrator uses the role templates provided by the developers for building role collections and assigning them to business users. The following diagram represents the different process steps for setting up the security artifacts.





## 4.2.1 Accessing SAP Leonardo IoT APIs Using OAuth Token

### Overview

After you subscribe your subaccount to SAP Leonardo IoT, you can retrieve the client secret from the service keys you created for your subaccount. For more information on retrieving the client secret, see [Retrieving ClientID and Client Secret](#). You can use the client secret to retrieve the OAuth token. The client secret is unique for a tenant and you can access all APIs and data within the tenant. This kind of authentication are used when you build backend applications, where you do not have a user context. You can automate configuring things, ingesting data, and retrieving data using the OAuth token.

### Retrieving OAuth Token using Postman

Postman is a popular and easy to use REST client using which you can access the SAP Leonardo IoT APIs. You can use the following API endpoint to retrieve the OAuth token specific for your tenant.

`https://<your tenant>.authentication.eu10.hana.ondemand.com/oauth/token`

Perform the following steps to retrieve the token:

1. Turn off the interceptor in Postman.
2. Send a POST request to the endpoint `https://<your tenant>.authentication.eu10.hana.ondemand.com/oauth/token`.
3. Enter the following in the **Headers** tab:

Key	Value
Content-Type	<code>application/x-www-form-urlencoded</code>

4. Enter the following in the **Body** tab:

Key	Value
grant_type	<code>client_credentials</code>
response_type	<code>token</code>
client_id	<code>sb-49fa22b5-4e44-46da-83eb-b5445c6c57e7!b15621</code>
client_secret	<code>iotae_service!b5</code>

The response contains the following values:

- **token:** Bearer token used to authenticate the requests
- **expiresIn:** Duration (in seconds) for which the token you received is valid
- **scope:** List of scopes required for executing a service

You can use this Bearer token with an API endpoint to retrieve data. To know more about the API endpoints to access the services offered by SAP Leonardo IoT, see [Application Names in Service URIs \[page 18\]](#). The table in the section **Application Names and Service URIs**, lists the service URIs with and without application router.

## Retrieving Data using Postman

The API endpoint to retrieve the package configuration data defined in your tenant using the **Service2Service** communication is `https://config-package-sap.cfapps.eu10.hana.ondemand.com/Package/v1/Packages`. The request is pointing directly to the microservice for thing configuration.

1. Send a POST request for the API endpoint to retrieve the package configuration.
2. Enter the following in the **Headers** tab:

Key	Value
Authorization	<code>Bearer &lt;&lt;your token&gt;&gt;</code>

The response contains the package configuration data.

### Note

You must be aware of the following when you call an API endpoint using the OAuth token:

- Only the most commonly used functional scopes for executing the services are hardcoded in the Bearer token. You must know that not the complete set of functional scopes are included in the token.
- The object instance-based authorization is not active. This means that the access rights to specific objects are not restricted. For example, the things created in a tenant is visible to all users who have

access to the tenant. Hence, the application calling the APIs using OAuth token must ensure to define their own filtering capabilities to restrict access to all available objects.

## Tutorial

[Call IoT Application Enablement service APIs with Postman](#) 

## 4.3 Role Templates and Scopes

Standard role templates provided by SAP for typical user profiles

### Overview

The big number of different scopes that are needed to grant (or revoke) functional authorizations in SAP Leonardo IoT can turn into a challenge for administrators who are in charge of assigning proper user profiles for the users of the system. To make this task easier, SAP provides a number of predefined role templates that an administrator may use as a blueprint for setting up individual roles as they are needed for a concrete organization. The basic procedure for this is as follows:

1. Derive a role from a role template.
2. Assign role (one or many) to a role collection.
3. Assign role collection (one or many) to a user group.

Role templates and scopes have a n:m relationship. That is, one template may contain many scopes, and one scope may be contained in many templates. The following sections show listings of the available role templates and the scopes that are contained in each template. The listings are broken down by the main areas of the SAP Leonardo IoT platform.

### Disclaimer

This section presents all the role templates and scopes supported to grant (or revoke) functional authorizations in SAP Leonardo IoT. When exploring the role templates, you may discover additional functional scopes that are not explained in this reference. These undocumented scopes are mostly providing authorization that are needed by the configuration services that are shipped by SAP as a part of the SAP Leonardo IoT.

## Business Partner

Role Templates for Business Partner

Role Template	Description	Scopes
<code>_Tenant_Administrator</code>	Perform all activities that are necessary for managing tenants as well as the data that is stored in a tenant.	<code>&lt;tenant&gt;.pers.c</code> <code>&lt;tenant&gt;.pers.r</code> <code>&lt;tenant&gt;.pers.u</code> <code>&lt;tenant&gt;.pers.d</code> <code>&lt;tenant&gt;.org.c</code> <code>&lt;tenant&gt;.org.r</code> <code>&lt;tenant&gt;.org.u</code> <code>&lt;tenant&gt;.org.d</code> <code>&lt;bp&gt;.tenant.r</code> <code>&lt;bp&gt;.tenant.u</code> <code>&lt;bp&gt;.c</code> <code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.u</code> <code>&lt;bp&gt;.d</code> <code>&lt;auth&gt;.c</code> <code>&lt;auth&gt;.r</code> <code>&lt;auth&gt;.u</code> <code>&lt;auth&gt;.d</code> <code>xs_authorization.read</code> <code>xs_authorization.write</code> <code>&lt;thing&gt;.TenantTSS.c</code> <code>&lt;thing&gt;.TenantTSS.d</code>
<code>_Person_Viewer</code>	View person data	<code>&lt;tenant&gt;.pers.r</code> <code>&lt;bp&gt;.r</code>

Role Template	Description	Scopes
<code>_Person_Editor</code>	Edit person data	<code>&lt;tenant&gt;.pers.c</code> <code>&lt;tenant&gt;.pers.r</code> <code>&lt;tenant&gt;.pers.u</code> <code>&lt;tenant&gt;.pers.d</code> <code>&lt;bp&gt;.c</code> <code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.u</code> <code>&lt;bp&gt;.d</code>
<code>_Organization_Viewer</code>	View organization data	<code>&lt;tenant&gt;.org.r</code> <code>&lt;bp&gt;.r</code>
<code>_Organization_Editor</code>	Edit organization data	<code>&lt;tenant&gt;.org.c</code> <code>&lt;tenant&gt;.org.r</code> <code>&lt;tenant&gt;.org.u</code> <code>&lt;tenant&gt;.org.d</code> <code>&lt;bp&gt;.c</code> <code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.u</code> <code>&lt;bp&gt;.d</code>
<code>_BusinessPartner_Viewer</code>	View person and organization data on elementary service level	<code>&lt;bp&gt;.r</code>
<code>_BusinessPartner_Editor</code>	Edit person and organization data on elementary service level	<code>&lt;bp&gt;.c</code> <code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.u</code> <code>&lt;bp&gt;.d</code>
<code>_BusinessPartner_OnboardTenant</code>	Perform all activities that are necessary for onboarding a new tenant	<code>&lt;bp&gt;.OnboardTenant</code>

## Authorization and Location

Role Templates for Authorization and Location

Role Template	Description	Scopes
<code>_Authorization_Viewer</code>	View authorization	<code>&lt;auth&gt;.r</code>
<code>_Authorization_Editor</code>	Edit authorization	<code>&lt;auth&gt;.c</code> <code>&lt;auth&gt;.r</code> <code>&lt;auth&gt;.u</code> <code>&lt;auth&gt;.d</code>
<code>_Location_Viewer</code>	View locations	<code>&lt;loc&gt;.r</code>
<code>_Location_Editor</code>	Edit locations	<code>&lt;loc&gt;.c</code> <code>&lt;loc&gt;.r</code> <code>&lt;loc&gt;.u</code> <code>&lt;loc&gt;.d</code>

## Thing

Role Templates for Thing

Role Template	Description	Scope
<code>Configuration_Viewer</code>	View configuration	<code>&lt;pkg&gt;.r</code> <code>&lt;thingconf&gt;.r</code>
<code>Configuration_Editor</code>	Create/Edit configuration	<code>&lt;pkg&gt;.c</code> <code>&lt;pkg&gt;.r</code> <code>&lt;pkg&gt;.u</code> <code>&lt;pkg&gt;.d</code> <code>&lt;thingconf&gt;.c</code> <code>&lt;thingconf&gt;.r</code> <code>&lt;thingconf&gt;.u</code> <code>&lt;thingconf&gt;.d</code>

Role Template	Description	Scope
Thing_Viewer	View thing	<thing>.r <thing>.Event.r <auth>.r <coldstore>.r <ohs>.r <ths>.r <advlist>.r <tg>.r <thngdtl>.r <loc>.r <geolocation>.r

Role Template	Description	Scope
Thing_Editor	Create/Edit thing	<thing>.c <thing>.r <thing>.d <thing>.Event.c <thing>.Event.r <thing>.Event.u <thing>.Event.d <auth>.c <auth>.r <auth>.u <auth>.d <coldstore>.c <coldstore>.r <coldstore>.d <ohs>.c <ohs>.r <ohs>.u <ohs>.d <ths>.c <ths>.r <ths>.u <ths>.d <tg>.c <tg>.r <tg>.u <tg>.d <bp>.tenant.r <bp>.vh.r <advlist>.r <thngdtl>.r <loc>.c



Role Template	Description	Scope
		<loc>.r
		<loc>.u
		<loc>.d
		<geolocation>.r
		<geolocation>.c
		<geolocation>.u

Role Template	Description	Scope
_Thing_Engineer	Comprises all authorizations needed by the Thing Modeler app	<thing>.c <thing>.r <thing>.d <thing>.Event.c <thing>.Event.r <thing>.Event.u <thing>.Event.d <auth>.r <conf>.c <conf>.r <conf>.d <tde>.c <tde>.r <tde>.d <bp>.conf.c <bp>.conf.r <bp>.conf.d <auth>.conf.c <auth>.conf.r <auth>.conf.d <ohs>.r <ohs>.c <ohs>.u <ohs>.d <ths>.r <ths>.c <ths>.d <ths>.u <tg>.c <tg>.r <tg>.u

Role Template	Description	Scope
		<tg>.d
		<tenant>.pers.r
		<advlist>.r
		<thngdtl>.r
		<bpanlyt>.r
		<pkg>.c
		<pkg>.r
		<pkg>.u
		<pkg>.d
		<thingconf>.c
		<thingconf>.r
		<thingconf>.u
		<thingconf>.d
		<loc>.c
		<loc>.r
		<loc>.u
		<loc>.d
		<file>.c
		<file>.r
		<file>.d
		<geolocation>.r
		<geolocation>.c
		<geolocation>.u
		<geolocation>.d

## Miscellaneous

Role Templates for Miscellaneous Tasks

Role Template	Description	Scope
_Monitoring	Check whether a service is running	<auth>.Uptime

Role Template	Description	Scope
_PhysicalDelete	Allows a user to see and physically delete logically deleted business partners as well as locations.	<bp>.pd <loc>.pd
_System_Admin	Perform system administration tasks like changing the log level	<auth>.SysAdmin <bp>.SysAdmin <tenant>.SysAdmin <loc>.SysAdmin <conf>.SysAdmin <thing>.SysAdmin <coldstore>.SysAdmin <ohs>.SysAdmin <ths>.SysAdmin
_ReuseUI_Viewer	Grants read access to a part of your data in SAP Leonardo IoT. This role provides all the authorizations required to test apps created with the SAP Leonardo IoT feature in SAP Web IDE and the available UI reuse controls.	<thing>.r <ct>.r <ctodata>.r <ct>.SysAdmin <ctodata>.SysAdmin <file>.r <ctodata>.SysAdmin <ctodata>.r <advlist>.r <thngdtl>.r <pkg>.r <thingconf>.r <ths>.r <tg>.r

## Related Information

[Functional Authorization \[page 263\]](#)

[UserGroup \[page 309\]](#)

# 5 Onboarding Services

Service mashups for complex operations during platform onboarding

## Prerequisites

To avoid confusion about the different aspects of the term "onboarding", we would like to start this section with the following clarification: Before you can start using the onboarding services described in the following chapters, you must have been onboarded as an SAP customer to SAP Cloud Platform. For more information, see the [Getting Started](#) guide for SAP Leonardo IoT.

## Introduction

In this section, you find services that consist of a combination of individual services used for managing the basic business partner or authorization entities of which the SAP Leonardo IoT platform consists. In addition, these compound services make it easier to map the functions and features of the identity provider to those of Cloud Foundry User Account and Authentication (UAA). The purpose of these compound services is to make it easier for a system administrator to set up all the necessary data for onboarding new persons, organizations, or tenants on to the platform. The most complex use case that is addressed by the compound services is tenant onboarding, where five different entities are involved plus some technical services.

### i Note

While this section is concentrating on onboarding organizations and persons, there is also another aspect of onboarding, namely onboarding of things. For this aspect, see the respective topic referred to in the *Related Information* section below.

The compound services that are listed here differ in the following respects from the services for the basic entities of the SAP Leonardo IoT platform:

- In almost all cases, the payload of the compound services is much bigger and more complex than the payload of the basic entities. This is because the compound services enable you to define the most relevant settings of several basic entities involved with one single call.
- Compared to the basic entities, the compound services come with a reduced set of methods on offer (for example, only a Create and a Delete method, but no method for updating). If you need to carry out an operation on an entity that was created with a compound service, you may use the complete set of methods that is available for the respective basic entity.

## Related Information

[Onboarding Organizations \[page 287\]](#)

[Onboarding Persons \[page 295\]](#)

[UserGroup \[page 309\]](#)

[IdentityZone \[page 318\]](#)

[Onboarding Things \[page 715\]](#)

## 5.1 Tenant

An entity used to separate the data of different customers of the platform owner from each other

A tenant is the top-level container object in the SAP Leonardo IoT architecture. It serves as a self-contained area where an administrator working for a Cloud Platform Provider sets up all the objects needed for a particular IoT scenario. As the business model of a cloud platform provider is aiming at onboarding a significant number of customers who want to use the services offered by the platform, it is very important to make sure that the data belonging to a particular customer is strictly separated from the data of all other customers using the same platform.

This separation of data is accomplished by the construct of the Tenant object. During the onboarding procedure, each customer is assigned to an individual tenant in the form of a 1:1 relationship. That is, there is no chance for a customer to access data that is not in the realm of its own tenant.

### i Note

The tenant service described here is the elementary service dedicated to tenant-related operations. It is mainly used internally by other services that belong to SAP Leonardo IoT. Due to the high degree of interrelationships with other services, we highly recommend using the mashup services instead rather than the elementary service. These mashups are all documented in the *Onboarding Services* section of this API reference documentation. Using this elementary Tenant service should be limited to read accesses.

### i Note

A tenant in SAP Leonardo IoT corresponds with a subaccount as implemented in the unified SAP Cloud Platform domain model.

### Tenant role and authorization

Depending on the role of a tenant to which a user is assigned, users are authorized to perform certain actions on the objects managed by the SAP Leonardo IoT platform:

- A user who is assigned to a tenant for a cloud platform provider (T0) can access objects that are assigned to a tenant assigned to a cloud platform user (T1). This is necessary because otherwise the platform provider would not be able to keep the platform services up and running. Also, it is a privilege of a T0 user to perform any kind of cross-tenant accesses. In other words, to perform cross-tenant activities, the object type in question must be prepared for allowing cross-tenant access **and** the user must be assigned to a tenant for a cloud platform provider (T0).
- A user who is assigned to a tenant for a cloud platform consumer (T1) can only access objects that are assigned to this particular tenant. There is no way for such a user to access objects that belong to other tenants. This is necessary for reasons of data protection.

## Naming Conventions for Packages

The `package` field is required and prefilled upon creation of a tenant. It is used for defining a namespace for the packages that will later be created within the scope of the tenant. The naming conventions for package namespaces are very similar (but **not** identical) to those in effect for the tenant itself:

- Allowed characters: [a..z, 0..9, .]
- Maximum length: 36 characters
- The first character must be a letter, that is, numbers are **not** allowed here.
- The namespace must consist of two segments, separated by a dot ("."). The first segment must have at least two characters, and the second segment must have at least three characters.
- Package namespaces have to be unique within one tenant. In addition, it is **not** allowed to define a namespace that is a substring of an already existing namespace.
- The namespace "com.sap" is reserved for SAP and cannot be used for customer-defined packages.

### Base URI:

- Formal description:
- Example for a base URI in a cloud foundry environment: `http://business-partner.cfapps.eu10.hana.ondemand.com/Tenants`

**Permissions:** The following permissions are needed for using the tenant services: `http://<server address>[:<port number>][</path>]/Tenants`

`<bp>.tenant.r`

### Note

The read permission for tenants is not only used in the context of the [Tenant](#) `http://` service itself. Rather, the `<bp>.tenant.r` permission is needed for successful execution of almost all the methods of the other services dealing with business partners and instance-based authorizations in the SAP Leonardo IoT platform. This is because the vast majority of the operations that can be performed in the platform are limited to the scope of the tenant where a user is logged on to. It is therefore crucial from a data protection perspective to ensure that each activity in the platform is performed within the boundaries of the proper tenant. As a consequence, the `<bp>.tenant.r` permission is needed for almost all service methods in addition to the service-specific permissions.

## Methods

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read a Tenant [page 284]</a>	<code>/Tenants ('&lt;tenantName&gt;')</code>	<code>&lt;bp&gt;.tenant.r</code>
GET	<a href="#">Read All Tenants [page 285]</a>	<code>/Tenants</code>	<code>&lt;bp&gt;.tenant.r</code>

## Payload

For PUT and GET, only the media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "tenant"      : { "type": "string", "maxLength": 36, "required": true },
  "etag"        : { "type": "string" },
  "organization": { "type": "string", "maxLength": 32},
  "package"     : { "type": "string", "maxLength": 40, "readOnly" : true,
"required": true }
}
```

## Related Information

[Package Configuration \[page 337\]](#)

### 5.1.1 Read a Tenant

Retrieves a specified tenant from the database

With this method, you send a request to the server to retrieve a particular Tenant object specified by its name.

If the tenant is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error.

## Request

URI:

HTTP Method: [GET](#)

Permissions: `Tenants('<tenant>')<bp>.tenant.r`

### Request Header Fields

None.

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
Tenant	Yes	string	Name of the tenant	Path

### Request Example

```
/Tenants('customer-test001')
```



## Response

### Response Status and Error Codes

Code	Reason
200	Tenant retrieved successfully

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "tenant"      : { "type": "string", "maxLength": 36, "required": true },
  "etag"        : { "type": "string" },
  "organization": { "type": "string", "maxLength": 32},
  "package"     : { "type": "string", "maxLength": 40, "readOnly" : true,
"required": true }
}
```

## Related Information

[Tenant \[page 282\]](#)

## 5.1.2 Read All Tenants

Retrieves a list of tenants from the database

With this method, you send a request to the server to retrieve a subset of tenants for which you are authorized.

### i Note

Due to a conceptual change in the architecture and authorization concept, this collective GET request always returns only the current tenant you are logged on to. Accessing objects outside of the scope of the current tenant is **not** supported. The service method has only been preserved for compatibility reasons. We recommend always using the single GET request as described in [Read a Tenant \[page 284\]](#).

## Request

URI: `/Tenants`

HTTP Method: [GET](#)

Permissions: <bp>.tenant.r

## Query String Parameters

Name	Required	Data Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set

### Note

Although from a technical perspective all the query parameters listed above are not mandatory, the requirement of restricting the result set of elements still remains in effect. It is only up to you to define a proper combination of parameters to fulfill this requirement.

## Request Example

```
/Tenants
```

Retrieves all tenants that are available in the database. Note that this unrestricted request is not supported due to unacceptable system load.

```
/Tenants?$top=25
```

Retrieves the first 25 tenants. Data is retrieved in page mode to reduce network load and response times.

```
/Tenants?$skip=10&$top=20
```

Retrieves 20 tenants from the database where the first 10 tenants are omitted. That is, the retrieved tenants are those with sequence numbers 11 through 30.

## Response

### Response Status and Error Codes

Code	Reason
200	Tenants retrieved successfully

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "tenant"      : { "type": "string", "maxLength": 36, "required": true },
  "etag"       : { "type": "string" },
  "organization": { "type": "string", "maxLength": 32},
}
```

```
"package"      : { "type": "string", "maxLength": 40, "readOnly" : true,
"required": true }
}
```

## Related Information

[Tenant \[page 282\]](#)

## 5.2 Onboarding Organizations

Describes services used for onboarding and offboarding an organization into a Leonardo IoT landscape

In the context of a Leonardo IoT landscape, the terms "onboarding" and "offboarding" are used in the following sense:

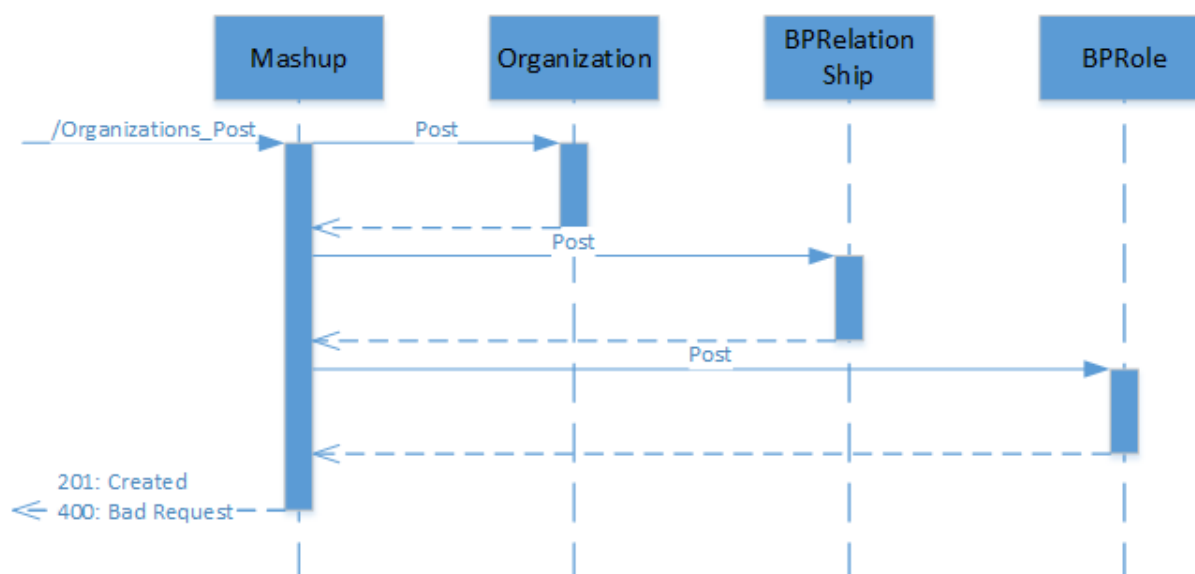
- **Onboarding:** The Cloud Platform Host that is responsible for operating the platform makes sure that an organization or a person is made known to the platform, maintains a set of master data for them, assigns them to an ObjectGroup defining their access rights within the platform. In case there is a business relationship between the newly created organization and an organization that already exists in the system, that relationship is also modeled in the data model of the platform.
- **Offboarding:** Whenever an organization does no longer require access to the services offered via the platform, it is up to the Cloud Platform Host to make sure that all relationships between this organization and other organizations are deleted, all access rights are revoked, and finally the organization is deleted from the system.

With these services, a business partner acting in the role of a Cloud Platform Provider can grant an organization access to the Leonardo IoT landscape as a Cloud Platform Consumer or as a Cloud Platform Consumer (2nd level). Likewise, the cloud platform provider can decide to revoke an access right already granted to an organization.

The structure of this service is very similar to the one described as [Organization \[page 28\]](#). The main differences are the following:

- The payload of the Onboarding services is a superset of the Organization payload. In addition, it comprises data sections used to indicate the assigned object group and the onboarding organization.
- An organization is either on board or not. Consequently, there is no need for the full CRUD method set that is offered for most of the other services. Creating and deleting the assignment of an organization to the platform is therefore sufficient.

The following interaction diagram illustrates the various steps that are processed during onboarding of an organization:



#### Base URI:

- Formal description: `http://<server address>[:<port number>]/[<path>]/Organizations`
- Example for a base URI in a cloud foundry environment: `http://tenant-administration.cfapps.neo.ondemand.com/Organizations`

**Permissions:** The following permissions are needed for using the organization onboarding services:

`<tenant>.org.c`

`<tenant>.org.r`

`<tenant>.org.d`

## Methods

HTTP Method	Action	URI	Scopes
<i>POST</i>	<a href="#">Create Organization Assignment [page 290]</a>	<code>/Organizations('&lt;businessPartnerID&gt;')</code>	<code>&lt;tenant&gt;.org.c</code> <code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.c</code> <code>&lt;bp&gt;.tenant.r</code>
<i>GET</i>	<a href="#">Read Organization Assignment [page 292]</a>	<code>/Organizations('&lt;businessPartnerID&gt;')</code>	<code>&lt;tenant&gt;.org.r</code> <code>&lt;bp&gt;.r</code> <code>&lt;bp&gt;.tenant.r</code>

HTTP Method	Action	URI	Scopes
DELETE	<a href="#">Delete Organization Assignment</a> <a href="#">[page 293]</a>	/ Organizations ('<businessPartnerID>')	<tenant>.org.d <bp>.r <bp>.d <bp>.tenant.r

## Payload

Only the media/type JSON is supported. The JSON for this method has the following structure:

```
{
  "organization": {
    "basicData": {
      "businessPartnerID": { "type": "string", "maxLength": 32 },
      "tenant": { "type": "string", "maxLength": 36 },
      "etag": { "type": "string" }
    },
    "organizationName": {
      "organizationName1": { "type": "string", "maxLength": 255, "required": true },
      "organizationName2": { "type": "string", "maxLength": 255 }
    },
    "communicationData": {
      "emailAddress": { "type": "string", "maxLength": 255, "required": true },
      "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
      "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
      "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
      "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
      "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
      "streetName": { "type": "string", "maxLength": 255 },
      "houseNumber": { "type": "string", "maxLength": 10 },
      "cityName": { "type": "string", "maxLength": 255 },
      "postalCode": { "type": "string", "maxLength": 10 },
      "country": { "type": "string", "maxLength": 2 },
      "countryDescription": { "type": "string", "maxLength": 60, "readOnly": true },
      "region": { "type": "string", "maxLength": 3 },
      "regionDescription": { "type": "string", "maxLength": 60, "readOnly": true }
    },
    "objectGroup": { "type": "string" }
  },
  "onboardingOrganization": {
    "businessPartnerID": { "type": "string", "maxLength": 32 }
  }
}
```

### Note

A special mechanism has been implemented for the fields dealing with phone numbers in the communication data block of the payload with respect to the preceding numeric country code. Unlike manual dialing with a phone, the country code is **not** given as the standardized number for a country (for example, 1 for USA, 49 for Germany). Instead, the system expects the alphabetical ISO country code as a prefix (for example, "US" for USA, "DE" for Germany). The following fields are affected:

- mobileDestinationLocation

- `landlineDestinationLocation`

This mechanism is in effect for the service request types POST, GET, and PUT. You may look up the required ISO country codes in the [Country Codes \[page 1244\]](#) section of this documentation.

## Related Information

[Business Partner \[page 25\]](#)

[Organization \[page 28\]](#)

[ObjectGroup \[page 206\]](#)

[BPRole \[page 102\]](#)

## 5.2.1 Create Organization Assignment

With this method, you create a new business partner of type organization and relate it to other entities in one step

Onboarding an organization consists of the following steps:

1. Creating a new ObjectGroup instance underneath the tenant root object group (only if no object group Id is provided).
2. Creating a new organization.
3. Creating a relationship between the onboarding organization and the newly created organization.
4. Creating a new BPRole instance of type `ROLE_CUSTOMER_OF_CUSTOMER`

Although assigning the new organization to an object group is an essential step during the onboarding process, it is still possible to create the new organization without providing an object group. In that case, the system creates a new object group with the tenant root object group as its parent object group. The new organization is then assigned to this object group.

## Request

**URI:** `/Organizations`

**HTTP Method:** `POST`

**Permissions:**

`<tenant>.org.c`

`<bp>.r`

`<bp>.c`

`<bp>.tenant.r`

## Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token

## Request Parameters

Name	Required	Type	Description	Send As
organizationName1	Yes	string	Semantic name of the organization	JSON
emailAddress	Yes	string	An email address used to get in contact with the organization.	JSON
tenant	No	string	If omitted, tenant of currently logged on user is used.	JSON

## Request Example

```
/Organizations
```

## Payload

Only the media/type JSON is supported. The JSON for this method has the following structure:

```
{
  "organization": {
    "basicData": {
      "businessPartnerID": { "type": "string", "maxLength": 32 },
      "tenant": { "type": "string", "maxLength": 36 },
      "etag": { "type": "string" }
    },
    "organizationName": {
      "organizationName1": { "type": "string", "maxLength": 255, "required": true },
      "organizationName2": { "type": "string", "maxLength": 255 }
    },
    "communicationData": {
      "emailAddress": { "type": "string", "maxLength": 255, "required": true },
      "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
      "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
      "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
      "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
      "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
      "streetName": { "type": "string", "maxLength": 255 },
      "houseNumber": { "type": "string", "maxLength": 10 },
      "cityName": { "type": "string", "maxLength": 255 },
      "postalCode": { "type": "string", "maxLength": 10 },
      "country": { "type": "string", "maxLength": 2 },
      "countryDescription": { "type": "string", "maxLength": 60, "readOnly": true },
      "region": { "type": "string", "maxLength": 3 },
      "regionDescription": { "type": "string", "maxLength": 60, "readOnly": true }
    },
    "objectGroup": { "type": "string" }
  },
  "onboardingOrganization": {
    "businessPartnerID": { "type": "string", "maxLength": 32 }
  }
}
```

```
}
```

## Response

Format: [JSON](#)

### Response Header Fields

Name	Type	Description
Location	string	Path to the newly created Organization
Etag	string	entity tag of the newly created Organization instance

### Response Status and Error Codes

Code	Reason
201	Organization created successfully
400	Bad request (for example, required field value missing)

## Related Information

[Onboarding Organizations \[page 287\]](#)

[ObjectGroup \[page 206\]](#)

[BPRole \[page 102\]](#)

## 5.2.2 Read Organization Assignment

Describes how to retrieve an onboarded organization and its related entities

With this method, you send a request to the server to retrieve a particular Organization object specified by its ID. Based on the retrieved organization instance, you can also access and modify its attributes as well as the attributes of the related entities, such as the onboarding organization, that is, the organization by which the current organization has been entered into the system.

## Request

URI: `/Organizations('<businessPartnerID>')`

HTTP Method: [GET](#)



#### Permissions:

<tenant>.org.r

<bp>.r

<bp>.tenant.r

#### Request Header Fields

Name	Required	Values
If-Match	No	Check if cached version matches server version of the organization

#### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPartnerID	Yes	string	Technical ID of the Organization object	Path

#### Request Example

```
/Organizations('d5fa4d8cf1c94c7b8b72c97b185d3711')
```

## Response

#### Response Status and Error Codes

Code	Reason
200	Organization retrieved successfully
404	Not found

## Related Information

[Onboarding Organizations \[page 287\]](#)

## 5.2.3 Delete Organization Assignment

With this method, you delete an Organization with the specified ID

With this method, you delete the organization with the specified ID from the system. If an organization is deleted, all assigned relationships to other business partners are deleted as well. The deletion includes also relations that have not been created using this service, for example, relations to other organizations. All other entities remain unchanged.

## i Note

Deleting an organization is done in the form of a logical deletion only. That is, from a user's perspective, the deleted organization is no longer visible and cannot be accessed by the user. However, the physical organization object in the database remains untouched and can be retrieved by administrators, if necessary. This can be important, for example, for auditing purposes.

If the organization is successfully deleted, the server sends HTTP response code 204. Any other code indicates an error.

## Request

**URI:** /Organizations ('<businessPartnerID>')

**HTTP Method:** *DELETE*

**Permissions:**

<tenant>.org.d

<bp>.r

<bp>.d

<bp>.tenant.r

## Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross-site request forgery token
If-Match	Yes	Check if cached version matches server version of the organization

## Request Parameters

Name	Required	Type	Description	Send As
businessPartnerID	Yes	string	technical ID of the Organization	Path parameter

## Request Example

```
/Organizations ('4459cb0648ec4c83a01b1fffd7bae205')
```

## Response

**Format:** JSON

## Response Status and Error Codes

Code	Reason
204	Organization deleted successfully
404	Not found (for example, due to a misspelled organization ID)

## Related Information

[Onboarding Organizations \[page 287\]](#)

## 5.3 Onboarding Persons

Describes Services used for onboarding and offboarding a person into a Leonardo IoT landscape.

With these services, a business partner acting in the role of a Cloud Platform Provider can grant a person access to the Leonardo IoT landscape and establish the person's relationships to other entities in the database. Likewise, the cloud platform provider can decide to revoke an access right already granted to a person. With that, this service aims at providing a single point of entry for maintaining person-related data and to keep them synchronized with the user data for that person.

A person can only be onboarded if there is an organization for which the person works as an employee. Onboarding a single person without a relationship to an organization is **not** possible. The employing organization also serves as a provider of access rights for the person, as the object group assigned to an organization is inherited by the organization's employees.

### Prerequisites

The following prerequisites must be fulfilled:

- The organization to which the person shall be assigned already exists.
- The tenant to which the onboarding organization belongs has access to the relevant SAP Cloud Identity data (SCI), for example the SCI endpoint. This is needed for the mapping between a person and the corresponding user as well as for accessing user groups.
- The following fields must be provided during creation of a person:
  - Person: Family name
  - Person: Given name
  - Person: Email address
  - Person: userActive
  - employingOrganization: ID

### Base URI:

- Formal description: `http://<server address>[:<port number>]/[<path>]/Persons`
- Example for a base URI in a cloud foundry environment: `http://tenant-administration.cfapps.neo.ondemand.com/Persons`

**Permissions:** The following permissions are needed for using the person onboarding services:

<tenant>.pers.c

<tenant>.pers.r

<tenant>.pers.u

<tenant>.pers.d

## Methods

In contrast to the onboarding services for organizations, onboarding for persons comes with the complete set of CRUD methods. This is because for a person, the service allows for additional features with respect to modifying an already established person assignment. For example, you can use this service to assign an already onboarded person to a different organization, or you can modify the complete set of data fields stored for a person.

HTTP Method	Action	URI	Scopes
POST	<a href="#">Create Person Assignment [page 298]</a>	/Persons	<tenant>.pers.c  <bp>.c  <bp>.tenant.r  <bp>.r
GET	<a href="#">Read Person Assignment [page 302]</a>	/Persons ('<businessPartnerID>')	<tenant>.pers.r  <bp>.r  <bp>.tenant.r
PUT	<a href="#">Update Person Assignment [page 304]</a>	/Persons ('<businessPartnerID>')	<tenant>.pers.u  <bp>.c  <bp>.r  <bp>.u  <bp>.d  <bp>.tenant.r
DELETE	<a href="#">Delete Person Assignment [page 307]</a>	/Persons ('<businessPartnerID>')	<tenant>.pers.d  <bp>.r  <bp>.tenant.r  <bp>.d

## Payload

Only the media/type JSON is supported. The JSON for these methods has the following structure:

```
{
  "person" : {
    "basicData" : {
      "businessPartnerID": { "type": "string", "maxLength": 32 },
      "tenant": { "type": "string", "maxLength": 36 },
      "etag": { "type": "string" }
    },
    "personName": {
      "formOfAddress": { "type": "string", "maxLength": 10 },
      "formOfAddressDescription": { "type": "string", "maxLength": 30, "readOnly" : true },
      "familyName": { "type": "string", "maxLength": 255, "required": true },
      "additionalFamilyName": { "type": "string", "maxLength": 255 },
      "familyNamePrefix": { "type": "string", "maxLength": 10 },
      "middleName": { "type": "string", "maxLength": 255 },
      "givenName": { "type": "string", "maxLength": 255, "required": true },
      "initials": { "type": "string", "maxLength": 10 },
      "academicTitle": { "type": "string", "maxLength": 4 },
      "academicTitleDescription": { "type": "string", "maxLength": 60, "readOnly" : true },
      "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
      "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60, "readOnly" : true }
    },
    "communicationData" : {
      "emailAddress": { "type": "string", "maxLength": 255, "required": true },
      "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
      "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
      "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
      "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
      "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
      "streetName": { "type": "string", "maxLength": 255 },
      "houseNumber": { "type": "string", "maxLength": 10 },
      "cityName": { "type": "string", "maxLength": 255 },
      "postalCode": { "type": "string", "maxLength": 10 },
      "country": { "type": "string", "maxLength": 2 },
      "countryDescription": { "type": "string", "maxLength": 60, "readOnly" : true },
      "region": { "type": "string", "maxLength": 3 },
      "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
    },
    "objectGroup" : { "type": "string" }
  },
  "userActive": { "type": "boolean" },
  "userGroups": {
    "value" : [
      {
        "name": { "type": "string", "maxLength": 30 }
      }
    ]
  },
  "employingOrganization" : {
    "businessPartnerID": { "type": "string", "maxLength": 32 }
  }
}
```

### Note

Although the `givenName` as well as the `familyName` fields are defined with a field length of 255 characters, this length can only be used partially if the Person shall be associated with a User object (which

applies in most cases). Due to limitations of the underlying user administration layer, the following restrictions apply:

- `givenName`: 31 characters
- `familyName`: 63 characters

### i Note

A special mechanism has been implemented for the fields dealing with phone numbers in the communication data block of the payload with respect to the preceding numeric country code. Unlike manual dialing with a phone, the country code is **not** given as the standardized number for a country (for example, 1 for USA, 49 for Germany). Instead, the system expects the alphabetical ISO country code as a prefix (for example, "US" for USA, "DE" for Germany). The following fields are affected:

- `mobileDestinationLocation`
- `landlineDestinationLocation`

This mechanism is in effect for the service request types POST, GET, and PUT. You may look up the required ISO country codes in the [Country Codes \[page 1244\]](#) section of this documentation.

## Related Information

[Business Partner \[page 25\]](#)

[Person \[page 55\]](#)

[ObjectGroup \[page 206\]](#)

## 5.3.1 Create Person Assignment

With this method, you create a new business partner of type person and relate it to other entities in one step

You use this method to create a person and to assign it to several other entities in the system. This method takes care of the following actions:

- Create a business partner object of type person.
- Assign the new person to an organization that acts as the person's employer (required).
- Relate the person to an object group describing the access rights for the person.
- Create a new user or map the person to an existing matching user for system access.
- Assign the person via its associated user to one or more usergroups (optional).

### Required Fields

The following fields must be provided during creation of a person:

- Person: Family name
- Person: Given name
- Person: Email address

- Person: userActive
- employingOrganization: ID

## Employing Organization

The employing organization is required and must exist in the system prior to onboarding a person. Its tenant determines the tenant to which the person is assigned. If no object group is provided, the system assigns the newly created person to the object group of the employing organization.

## Relation between Person and User

A person can have either one user for accessing the system or none. The relation between person and user is controlled via the `<activeUser>` field in the payload of this service. Here, the following scenarios can occur during creation:

- User with matching name and email address already exists: The system associates the newly created person with the already existing user.
- User with matching name and email address does **not** exist: The system creates a new user with a subset of the person's data and associates person and user.
- User with either matching name or matching email address exists, but not both: The system returns an error, and the intended relation between person and user is **not** established.

When the system creates a new user for a person, the following field values are automatically transferred to the user object:

- Family name
- Email address

Currently, the following limitations apply with respect to the person onboarding service:

- If the last name of a person changes, it is currently **not** possible to reflect that change in the corresponding user data automatically.
- Data synchronization between person and user is possible only from person to user, but not vice versa. That is, if user data is changed directly by calling the respective SAP Cloud Identity (SCI) services or via the SCI administration console, these changes are **not** reflected in the corresponding person data.
- The email address of a business partner (person or organization) must be unique. Creating a business partner with an already existing email address is not supported. The same applies for changing the email address to a value already used in another business partner.

### i Note

There is one exception to this rule: It is possible to use the same email address for one person **and** one organization within a tenant. This allows for defining a generic, non-individual email address like `support@company.com` and assigning it to both a person and to the organization the person belongs to.

- Also, the email address of a user must be unique.

## Request

URI: `/Persons`

HTTP Method: `POST`

## Permissions:

<tenant>.pers.c

<bp>.c

<bp>.tenant.r

<bp>.r

## Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross site request forgery token

## Request Fields

Parameter	Required	Data Type	Description	Parameter Type
tenant	Yes	string	If omitted, tenant of currently logged on user is used.	JSON
familyName	Yes	string	Last name of the person to be created.	JSON
givenName	Yes	string	First name of the person to be created.	JSON
emailAddress	Yes	string	An email address used to get in contact with the person.	JSON

## Request Example

```
/Persons
```

## Payload

Only the media/type JSON is supported. The JSON for this method has the following structure:

```
{ "person" :
  { "basicData" : {
    "businessPartnerID": { "type": "string", "maxLength": 32 },
    "tenant": { "type": "string", "maxLength": 36 },
    "etag": { "type": "string" }
  },
  "personName": {
    "formOfAddress": { "type": "string", "maxLength": 10 },
    "formOfAddressDescription": { "type": "string", "maxLength": 30, "readOnly" :
true },
    "familyName": { "type": "string", "maxLength": 255, "required": true },
    "additionalFamilyName": { "type": "string", "maxLength": 255 },
    "familyNamePrefix": { "type": "string", "maxLength": 10 },
    "middleName": { "type": "string", "maxLength": 255 },
    "givenName": { "type": "string", "maxLength": 255, "required": true },
    "initials": { "type": "string", "maxLength": 10 },
    "academicTitle": { "type": "string", "maxLength": 4 },
    "academicTitleDescription": { "type": "string", "maxLength": 60, "readOnly" :
true },
  }
}
```



```

    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,
"readOnly" : true }
},
"communicationData" : {
    "emailAddress": { "type": "string", "maxLength": 255, "required": true },
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
    "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
    "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
    "streetName": { "type": "string", "maxLength": 255 },
    "houseNumber": { "type": "string", "maxLength": 10 },
    "cityName": { "type": "string", "maxLength": 255 },
    "postalCode": { "type": "string", "maxLength": 10 },
    "country": { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60, "readOnly" : true },
    "region": { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
},
"objectGroup" : { "type": "string"
},
"userActive": { "type": "boolean" },
"userGroups":{
    "value" : [
        {
            "name": { "type": "string", "maxLength": 30 }
        }
    ],
"employingOrganization" : {
    "businessPartnerID": { "type": "string", "maxLength": 32 }
}
}
}

```

## i Note

Although the `givenName` as well as the `familyName` fields are defined with a field length of 255 characters, this length can only be used partially if the Person shall be associated with a User object (which applies in most cases). Due to limitations of the underlying user administration layer, the following restrictions apply:

- `givenName`: 31 characters
- `familyName`: 63 characters

## Response

### Response Header Fields

Name	Description
Location	Path to the newly created Person assignment
ETag	Entity tag of the newly created Person assignment

## Response Status and Error Codes

Code	Reason
201	Person assignment created successfully
400	Bad request

## Related Information

[Onboarding Persons \[page 295\]](#)

[Person \[page 55\]](#)

## 5.3.2 Read Person Assignment

Describes how to retrieve an onboarded person and its related entities

With this method, you send a request to the server to retrieve a particular Person object specified by its ID. Based on the retrieved person instance, you can also access and modify its attributes as well as the attributes of the related entities, such as the employing organization.

If the person is successfully retrieved, the server sends HTTP response code 200. Any other code indicates an error

## Request

**URI:** `/Persons('<businessPartnerID>')`

**HTTP Method:** *GET*

**Permissions:**

`<tenant>.pers.r`

`<bp>.r`

`<bp>.tenant.r`

## Request Header Fields

Header	Required	Description
If-Match	Yes	Check if cached version matches server version of the person

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPartnerID	Yes	string	Technical ID of the Person object	Path

## Request Example

```
/Persons('5d9616a522a64ed6974b01e4468cf8b1')
```

## Response

### Response Status and Error Codes

Code	Reason
200	Person retrieved successfully
404	Not found

## Payload

Only the media/type JSON is supported. The JSON for this method has the following structure:

```
{ "person" :  
  { "basicData" : {  
    "businessPartnerID": { "type": "string", "maxLength": 32 },  
    "tenant": { "type": "string", "maxLength": 36 },  
    "etag": { "type": "string" }  
  },  
  "personName": {  
    "formOfAddress": { "type": "string", "maxLength": 10 },  
    "formOfAddressDescription": { "type": "string", "maxLength": 30, "readOnly" :  
true },  
    "familyName": { "type": "string", "maxLength": 255, "required": true },  
    "additionalFamilyName": { "type": "string", "maxLength": 255 },  
    "familyNamePrefix": { "type": "string", "maxLength": 10 },  
    "middleName": { "type": "string", "maxLength": 255 },  
    "givenName": { "type": "string", "maxLength": 255, "required": true },  
    "initials": { "type": "string", "maxLength": 10 },  
    "academicTitle": { "type": "string", "maxLength": 4 },  
    "academicTitleDescription": { "type": "string", "maxLength": 60, "readOnly" :  
true },  
    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },  
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,  
"readOnly" : true }  
  },  
  "communicationData" : {  
    "emailAddress": { "type": "string", "maxLength": 255, "required": true },  
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },  
    "mobilePhoneNumber": { "type": "string", "maxLength": 127 },  
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },  
    "landlinePhoneNumber": { "type": "string", "maxLength": 127 },  
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },  
    "streetName": { "type": "string", "maxLength": 255 },  
  }  
}
```

```

"houseNumber": { "type": "string", "maxLength": 10 },
"cityName": { "type": "string", "maxLength": 255 },
"postalCode": { "type": "string", "maxLength": 10 },
"country": { "type": "string", "maxLength": 2 },
"countryDescription": { "type": "string", "maxLength": 60, "readOnly" : true },
"region": { "type": "string", "maxLength": 3 },
"regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
},
"objectGroup" : { "type": "string"
},
"userActive": { "type": "boolean" },
"userGroups":{
  "value" : [
    {
      "name": { "type": "string", "maxLength": 30 }
    }
  ],
},
"employingOrganization" : {
  "businessPartnerID": { "type": "string", "maxLength": 32 }
}
}

```

## Related Information

[Onboarding Persons \[page 295\]](#)

[Create a Person \[page 59\]](#)

## 5.3.3 Update Person Assignment

With this method, you modify the person assignment of the person with the specified ID

In the payload of the request, you assign values to those fields that you want to modify. For example, to add the birth name of a female person in addition to the mandatory `<familyName>`, you may pass a value for `<additionalFamilyName>` in the payload of the service request. With that, you could differentiate between different persons who, at first sight, have the same name. The only limitation here is that you cannot change those attributes that are needed for technically identifying the person (that is, `<tenant>`, `<businessPartnerID>`, and `<etag>`)

With this onboarding service method, you cannot only modify the fields that directly belong to the onboarded person but also those fields belonging to the related entities that are currently associated with that person. This includes changing the references to associated entities. In other words, you can, for example, reassign a person to a different organization or object group.

### Relation between Person and User

The relation between an onboarded person and the user assigned to that person is controlled via the `<userActive>` flag. Toggling this flag during the Update service has the following effects:

- Set `<userActive>` to `true`  
 Either a new user object is created, or an existing but currently unassigned user with matching data is detected by the system. In both cases, the system automatically associates the person with the user. If the system creates a new user, the user fields are populated with the corresponding data of the person. For more information, see [Create Person Assignment \[page 298\]](#).

- Set `<userActive>` to `false`

The user that was previously associated with the person is set to inactive and can no longer be used for logging on to the system unless it is reassigned to a person at a later point in time. All other user data remains unchanged. As a result, the person in question is still known in the system, but cannot access the system any longer.

## Request

**URI:** `/Persons(<businessPartnerID>)`

**HTTP Method:** `PUT`

**Permissions:**

`<tenant>.pers.u`

`<bp>.c`

`<bp>.r`

`<bp>.u`

`<bp>.d`

`<bp>.tenant.r`

### Request Header Fields

Name	Required	Values
If-Match	Yes	Check if cached version matches server version of the person
X-CSRF-Token	Yes	Cross site request forgery token

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPartnerID	Yes	string	ID of the person	Path parameter

### Request Example

```
/Persons('5d9616a522a64ed6974b01e4468cf8b1')
```

## Payload

Only the media type JSON is supported. The JSON for this method has the following structure:

```
{ "person" :
  { "basicData" : {
    "businessPartnerID": { "type": "string", "maxLength": 32 },
```

```

    "tenant": { "type": "string", "maxLength": 36 },
    "etag": { "type": "string" }
  },
  "personName": {
    "formOfAddress": { "type": "string", "maxLength": 10 },
    "formOfAddressDescription": { "type": "string", "maxLength": 30, "readOnly" :
true },
    "familyName": { "type": "string", "maxLength": 255, "required": true },
    "additionalFamilyName": { "type": "string", "maxLength": 255 },
    "familyNamePrefix": { "type": "string", "maxLength": 10 },
    "middleName": { "type": "string", "maxLength": 255 },
    "givenName": { "type": "string", "maxLength": 255, "required": true },
    "initials": { "type": "string", "maxLength": 10 },
    "academicTitle": { "type": "string", "maxLength": 4 },
    "academicTitleDescription": { "type": "string", "maxLength": 60, "readOnly" :
true },
    "additionalAcademicTitle": { "type": "string", "maxLength": 4 },
    "additionalAcademicTitleDescription" : { "type": "string", "maxLength": 60,
"readOnly" : true }
  },
  "communicationData" : {
    "emailAddress": { "type": "string", "maxLength": 255, "required": true },
    "mobileDestinationLocation": { "type": "string", "maxLength": 2 },
    "mobilePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlineDestinationLocation": { "type": "string", "maxLength": 2 },
    "landlinePhoneNumber": { "type": "string", "maxLength": 127 },
    "landlinePhoneNumberExtension": { "type": "string", "maxLength": 127 },
    "streetName": { "type": "string", "maxLength": 255 },
    "houseNumber": { "type": "string", "maxLength": 10 },
    "cityName": { "type": "string", "maxLength": 255 },
    "postalCode": { "type": "string", "maxLength": 10 },
    "country": { "type": "string", "maxLength": 2 },
    "countryDescription": { "type": "string", "maxLength": 60, "readOnly" : true },
    "region": { "type": "string", "maxLength": 3 },
    "regionDescription": { "type": "string", "maxLength": 60, "readOnly" : true }
  },
  "objectGroup" : { "type": "string"
},
  "userActive": { "type": "boolean" },
  "userGroups":{
    "value" : [
      {
        "name": { "type": "string", "maxLength": 30 }
      }
    ],
  },
  "employingOrganization" : {
    "businessPartnerID": { "type": "string", "maxLength": 32 }
  }
}

```

## i Note

Although the `givenName` as well as the `familyName` fields are defined with a field length of 255 characters, this length can only be used partially if the Person shall be associated with a User object (which applies in most cases). Due to limitations of the underlying user administration layer, the following restrictions apply:

- `givenName`: 31 characters
- `familyName`: 63 characters

## Response

### Response Header Fields

Name	Description
Etag	Identifies the updated Person instance

### Response Status and Error Codes

Code	Reason
200	Person updated successfully
400	Bad request
404	Not found

## Related Information

[Onboarding Persons \[page 295\]](#)

[Create Person Assignment \[page 298\]](#)

## 5.3.4 Delete Person Assignment

With this method, you delete a Person assignment with the specified ID

With this method, you delete the person with the specified ID from the system. If a person is deleted, all assigned relationships to other entities (for example, business partners or object groups) are deleted as well. If the person is associated with a user, that user is automatically deactivated.

### **i** Note

Deleting a person is done in the form of a logical deletion only. That is, from a user's perspective, the deleted person is no longer visible and cannot be accessed by the user. However, the physical Person object in the database remains untouched and can be retrieved by administrators, if necessary. This can be important, for example, for auditing purposes.

As opposed to the Person object, all the relationships of the person to other entities are deleted physically. This is true for all relationships where the person in question is involved, regardless of whether these relations have been established with the help of the onboarding service or not.

While the person's relationships to other entities are deleted, the objects that were related to the person remain untouched by this service. They can therefore still be used for other purposes (for example, attaching them to other persons).

## Request

**URI:** /Persons ('<businessPartnerID>')

**HTTP Method:** *DELETE*

**Permissions:**

<tenant>.pers.d

<bp>.r

<bp>.tenant.r

<bp>.d

### Request Header Fields

Name	Required	Description
X-CSRF-Token	Yes	Cross site request forgery token
If-Match	Yes	Check if cached version matches server version of the person.

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
businessPartnerID	Yes	string	technical ID of the Person object	Path

### Request Example

```
/Persons ('5d9616a522a64ed6974b01e4468cf8b1')
```

## Response

### Response Status and Error Codes

Code	Reason
204	Person deleted successfully
404	Not found

## Related Information

[Onboarding Persons \[page 295\]](#)



## 5.4 UserGroup

A UserGroup serves as a connector between a user account and access rights.

### Note

Due to a change in the tenant administration concept, the User Groups app is no longer needed for newly created tenants. Use this service only if you have to access legacy tenants that have been set up with an older release of SAP Leonardo IoT.

A UserGroup object contains references to a list of ObjectGroupCapabilities as well as to a list of ObjectGroupRoleCollections defined for a specified tenant. In a later step, the UserGroup can be assigned to a user in the SAP Leonardo IoT landscape. As a consequence, the user receives all the access rights and authorizations defined by the list objects contained in the UserGroup. The ID of a UserGroup must be unique within the scope of the tenant to which it belongs.

### Base URI:

- Formal description: `http://<server address>[:<port number>][<path>]/UserGroups?$filter=<filter condition>`
- Example for a base URI in a cloud foundry environment: `http://tenant-administration.cfapps.neo.ondemand.com/UserGroups?$filter=tenant eq {{TenantT1_migrated}}&skipCapabilities=true`

## Methods

HTTP Method	Action	URI	Scopes
POST	<a href="#">Create a UserGroup [page 311]</a>	/UserGroups	<tenant>.pers.c <auth>.r <auth>.c <bp>.tenant.r xs_authorization.read xs_authorization.write
GET	<a href="#">Read a UserGroup [page 313]</a>	/UserGroups (<ID>)	<tenant>.pers.r <auth>.r <bp>.tenant.r xs-authorization.read

HTTP Method	Action	URI	Scopes
<i>GET</i>	<a href="#">Read all UserGroups [page 314]</a>	/UserGroups	<tenant>.pers.r <auth>.r <bp>.tenant.r xs-authorization.read
<i>PUT</i>	<a href="#">Update a UserGroup [page 315]</a>	/UserGroups (<ID>)	<tenant>.pers.u <auth>.r <auth>.c <auth>.u <auth>.d <bp>.tenant.r xs-authorization.read xs- authorization.write
<i>DELETE</i>	<a href="#">Delete a UserGroup [page 317]</a>	/UserGroups (<ID>)	<tenant>.pers.d <auth>.r <auth>.d <bp>.tenant.r xs-authorization.read xs- authorization.write

### Note

When you are calling the methods of the UserGroup service, you do not only need the permissions provided by the SAP Leonardo IoT framework. In addition, the respective permissions of the Cloud Foundry User Account and Authentication (UAA) system are required. These permissions can be recognized by their naming convention `xs-authorization.<access type>`.

## Payload

Only media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "id"          : { "type": "string", "maxLength": 32 },
  "tenant"      : { "type": "string", "maxLength": 36 },
  "name"        : { "type": "string", "maxLength": 30 },
```

```

"display_name" : { "type": "string", "maxLength": 30 },
"description"  : { "type": "string", "maxLength": 30 },
"objectGroupRoleCollections" : {
  "value": {
    [
      "name"      : { "type": "string", "maxLength": 32 }
    ]
  }
}
}

```

## Related Information

[Business Partner \[page 25\]](#)

### 5.4.1 Create a UserGroup

With this method, you create a new UserGroup object and relate it to other entities in one step

Setting up a user group consists of the following steps, which are performed automatically during execution of this method:

1. Creating a new UserGroup object.
2. Creating and assigning ObjectGroupCapabilities for the UserGroup.
3. Assigning role collections to the UserGroup.

## Request

**URI:** /UserGroups

**HTTP Method:** *POST*

**Permissions:**

<tenant>.pers.c

<auth>.r

<auth>.c

<bp>.tenant.r

xs\_authorization.read

xs\_authorization.write

## Request Header Fields

None.

## Request Parameters

None.

## Request Example

```
/UserGroups
```

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "id"          : { "type": "string", "maxLength": 32 },
  "tenant"      : { "type": "string", "maxLength": 36 },
  "name"        : { "type": "string", "maxLength": 30 },
  "display_name": { "type": "string", "maxLength": 30 },
  "description" : { "type": "string", "maxLength": 30 },
  "objectGroupRoleCollections" : {
    "value": {
      [
        "name" : { "type": "string", "maxLength": 32 }
      ]
    }
  }
}
```

## Response

### Response Headers

Name	Description
Location	Path to the newly created UserGroup

### Response Status and Error Codes

Code	Reason
201	UserGroup created successfully

## Related Information

[UserGroup \[page 309\]](#)

## 5.4.2 Read a UserGroup

Retrieves a specified UserGroup from the database

With this method, you send a request to the server to retrieve a particular UserGroup object specified by its name and the tenant name. In addition to the UserGroup object, the system also returns the role collections as well as the object group capabilities that are assigned to the user group.

### Request

**URI:** /UserGroups (name=<name>, tenant=<tenant>)

**HTTP Method:** [GET](#)

**Permissions:**

<tenant>.pers.r

<auth>.r

<bp>.tenant.r

xs-authorization.read

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
name	Yes	string	Name of the UserGroup	Path
tenant	Yes	string	Name of the tenant	Path

### Request Example

```
/UserGroups (name=UG1, tenant=ab-customer001)
```

### Response

#### Response Status and Error Codes

Code	Reason
200	UserGroup retrieved successfully

### Related Information

[UserGroup \[page 309\]](#)

## 5.4.3 Read all UserGroups

Retrieves a list of UserGroups from the database

With this method, you send a request to the server to retrieve a subset of all UserGroups, according to the filter criteria provided. For a collective request for a set of UserGroups, you have these options:

- You can restrict the set of user groups to be retrieved by filter criteria based on the tenant. In fact, defining filter criteria is mandatory. The system does not accept an unrestricted request for a list of all user groups in the database.
- You can define that the set of matching user groups shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

### Request

**URI:** /UserGroups

**HTTP Method:** *GET*

**Permissions:**

<tenant>.pers.r

<auth>.r

<bp>.tenant.r

xs-authorization.read

### Query String Parameters

Parameter	Required	Data Type	Description
\$filter	Yes	string	Filter condition to be applied; valid field is tenant

### Request Example

```
/UserGroups?$filter=tenant eq 'customer-test001'
```

Retrieves all user groups that have been defined for the specified tenant.

### Response

#### Response Status and Error Codes

Code	Reason
200	UserGroups retrieved successfully

## Related Information

[UserGroup \[page 309\]](#)

### 5.4.4 Update a UserGroup

With this method, you modify a user group with the specified name

In the payload of the request, you assign values to those fields that you want to modify. Note that you cannot change the elementary fields of the UserGroup (like `<display_name>` or `<description>`). If you need to change these fields, you can accomplish this via the SAP Cloud Identity console. However, you can replace the assigned `ObjectGroupCapabilities` and `ObjectGroupRoleCollections`.

#### i Note

During execution of this method, the system **always** deletes the existing assignments of both `ObjectGroupCapabilities` as well as `ObjectGroupRoleCollections`. It is therefore important to take extra care of providing the required assignments every time you call this method, even if you want to keep all existing assignments unchanged.

## Request

**URI:** `/UserGroups (name=<name>, tenant=<tenant>)`

**HTTP Method:** *PUT*

**Permissions:**

`<tenant>.pers.u`

`<auth>.r`

`<auth>.c`

`<auth>.u`

`<auth>.d`

`<bp>.tenant.r`

`xs-authorization.read`

`xs-authorization.write`

## Request Header Fields

Name	Required	Values
X-CSRF-Token	Yes	Cross site request forgery token

## Request Parameters

Name	Required	Data Type	Description	Parameter Type
name	Yes	string	Name of the UserGroup	Path

## Request Example

```
/UserGroups(name=UG01, tenant=ab-customer001)
```

## Payload

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "id"          : { "type": "string", "maxLength": 32 },
  "tenant"      : { "type": "string", "maxLength": 36 },
  "name"        : { "type": "string", "maxLength": 30 },
  "display_name": { "type": "string", "maxLength": 30 },
  "description" : { "type": "string", "maxLength": 30 },
  "objectGroupRoleCollections" : {
    "value": {
      [
        "name" : { "type": "string", "maxLength": 32 }
      ]
    }
  }
}
```

## Response

### Response Status and Error Codes

Code	Reason
200	UserGroup updated successfully
400	Bad request (for example, if you try to update one of the elementary fields)

## Related Information

[UserGroup \[page 309\]](#)



## 5.4.5 Delete a UserGroup

With this method, you delete a UserGroup with the specified name.

Calling the Delete method deletes the UserGroup instance, and it also erases the existing references to ObjectGroupCapabilities as well as to ObjectGroupRoleCollections of the UserGroup.

### Request

**URI:** /UserGroups (name=<name>, tenant=<tenant>)

**HTTP Method:** *DELETE*

**Permissions:**

<tenant>.pers.d

<auth>.r

<auth>.d

<bp>.tenant.r

xs-authorization.read

xs-authorization.write

### Request Header Fields

Name	Required	Values
X-CSRF-Token	Yes	Cross site request forgery token

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
name	Yes	string	Name of the UserGroup	Path

### Request Example

```
/UserGroups (name=UG01, tenant=ab-customer001)
```

### Response

#### Response Status and Error Codes

Code	Reason
204	UserGroup deleted successfully

## Related Information

[UserGroup \[page 309\]](#)

## 5.5 IdentityZone

A logical structure used to manage the connection between Cloud Foundry User Account and Authentication (UAA) and an identity provider service like SAP Cloud Identity (SCI).

You use this service to set up a user group with administrator authorizations for the SAP Leonardo IoT landscape. For this, during execution of the POST method, you pass the credentials of one administrative user account each belonging to the two platforms involved:

- Cloud Foundry User Account and Authentication (UAA)
- An identity provider service like SAP Cloud Identity (SCI)

Once logged on to the two platforms, a trusted relationship is established between the two platforms, which is required by the tenant onboarding service for setting up a new tenant.

### Base URI:

- Formal description: `http://<server address>[:<port number>]/[<path>]/IdentityZone`
- Example for a base URI in a cloud foundry environment: `http://tenant-administration.cfapps.neo.ondemand.com/IdentityZone`

**Permissions:** `<tenant>.tenant.c`

## Methods

HTTP Method	Action	URI	Scopes
<i>POST</i>	<a href="#">Create an IdentityZone [page 319]</a>	/IdentityZone	<tenant>.tenant.c

## Payload

Only the media type `application/json` is supported. The JSON for this method has the following structure:

```
{
  "Name"           : { "type": "string", "maxLength": 32, "required": true},
  "Description"    : { "type": "string", "maxLength": 255},
  "uaaURI"         : { "type": "string", "maxLength": 255, "required": true},
  "uaaUser"        : { "type": "string", "maxLength": 50, "required": true},
  "uaaPassword"    : { "type": "string", "maxLength": 50, "required": true},
  "idpURI"         : { "type": "string", "maxLength": 255, "required": true},
  "idpUser"        : { "type": "string", "maxLength": 50, "required": true},
```

```
"idpPassword"      : { "type": "string", "maxLength": 50, "required": true},
"idpCert"          : { "type": "string" },
"UserGroup"        : { "type": "string", "required": true}
}
```

#### Field Explanation

Name	Description
Name	Name of the IdentityZone
Description	Explanation of the purpose or specific boundary conditions of the IdentityZone
uaaURI, uaaUser, uaaPassword	Credentials of an administrator user defined for the Cloud Foundry User Account and Authentication (UAA) environment
idpURI, idpUser, idpPassword	Credentials of an administrator user defined for an identity provider service like SAP Cloud Identity (SCI)
idpCert	Certificate string for certificate-based logon to the identity provider service
UserGroup	Name of the new user group to be created

## Related Information

[UserGroup \[page 309\]](#)

## 5.5.1 Create an IdentityZone

Describes how to set up a new identity zone.

With this method, you set up a new identity zone that serves as a trusted environment for both the Leonardo IoT landscape and an external identity provider service. After the trusted relationship has been established, the service creates a new user group. Users assigned to that user group are authorized to manage objects stored in the Leonardo IoT platform with administrator rights.

### Prerequisites

Prior to creating a new identity zone, the following requirements must be fulfilled:

- An OAuth2 Cloud Foundry (CF) User Account and Authentication (UAA) administrator client and password must be provided in the service payload.
- A user with administrator authorization has been defined via the corresponding identity provider (IDP; for example, SAP Cloud Identity).

- The following fields must be provided during creation of a tenant:
  - A short Name of the new identity zone.
  - uaaURI
  - uaaUser
  - uaaPassword
  - idpURI
  - idpUser
  - idpPassword
  - UserGroup

With this method, the following services are created:

- UAA IdentityZone
- UAA Role Collection
- IDP Service Provider

During execution, metadata are exchanged between UAA and IDP. This results in a trusted relationship. The created UAA role collection is linked to the provided UserGroup and to the BOOTSTRAP RoleTemplate of this application. This will enable all users within this UserGroup to use the Onboarding Services as an administrator.

## Request

**URI:** /IdentityZone

**HTTP Method:** *POST*

**Permissions:** <tenant>.tenant.c

### Request Header Fields

Name	Required	Values
X-CSRF-Token	Yes	Cross-site request forgery token

### Request Example

```
/IdentityZone
```

## Payload

Only the media type `application/json` is supported. The JSON for this method has the following structure:

```
{
  "Name"           : { "type": "string", "maxLength": 32, "required": true},
  "Description"    : { "type": "string", "maxLength": 255},
  "uaaURI"         : { "type": "string", "maxLength": 255, "required": true},
  "uaaUser"        : { "type": "string", "maxLength": 50, "required": true},
  "uaaPassword"    : { "type": "string", "maxLength": 50, "required": true},
  "idpURI"         : { "type": "string", "maxLength": 255, "required": true},
```

```

    "idpUser"      : { "type": "string", "maxLength": 50, "required": true},
    "idpPassword"  : { "type": "string", "maxLength": 50, "required": true},
    "idpCert"      : { "type": "string" },
    "UserGroup"    : { "type": "string", "required": true}
  }

```

#### Field Explanation

Name	Description
Name	Name of the IdentityZone
Description	Explanation of the purpose or specific boundary conditions of the IdentityZone
uaaURI, uaaUser, uaaPassword	Credentials of an administrator user defined for the Cloud Foundry User Account and Authentication (UAA) environment
idpURI, idpUser, idpPassword	Credentials of an administrator user defined for an identity provider service like SAP Cloud Identity (SCI)
idpCert	Certificate string for certificate-based logon to the identity provider service
UserGroup	Name of the new user group to be created

## Response

### Response Status and Error Codes

Code	Reason
201	IdentityZone created successfully
400	Bad request

## Related Information

[IdentityZone \[page 318\]](#)

## 6 Data Protection and Privacy

General Data Protection Regulation Manager (GDPR) is an information framework that deals with Privacy of Personal Data. This service can be used by Data subject to identify and handle the privacy of personal data in the system. SAP provides specific features and functions to support compliance with regards to relevant legal requirements, including data protection. SAP Leonardo IoT delivers predefined annotations `com.sap.appiot.security:pii` and `com.sap.appiot.security:spi` to support compliance with regard to relevant legal requirements, including data protection.

You can assign the predefined annotations for property set types when you create a property set type or update a property set type.

After you define the annotations, all properties of the property set type are considered as personal data or sensitive data based on the annotation defined. You can remove or add the annotation for the property set type, only if data does not exist for the properties of the property set type. Whenever you create or modify data for properties that are identified as personal data or sensitive data, the system must generate audit logs.

### Note

You cannot assign the predefined annotations for property set types of data category **ReferencePropertyData**. If the referred property set type is considered as personal information or sensitive personal information by assigning the predefined annotations (`com.sap.appiot.security:pii` and `com.sap.appiot.security:spi`), the reference property data such as threshold values, unit of measure, and target values are also treated as personal information or sensitive personal information.

Assigning the `com.sap.appiot.security:pii` (personal information) annotation to a property set type can lead to decreased performance for data ingestion and writing data for property sets derived from that property set type.

Assigning the `com.sap.appiot.security:spi` (sensitive personal information) annotation to a property set type can lead to decreased read performance for property sets derived from that property set type.

The following are examples of various configuration payloads illustrating how predefined annotations are assigned for a property set type.

### Property Set Type Configuration (OData Service)

In the following example payload for creating a property set type, the predefined annotation `com.sap.appiot.security:spi` is assigned to the property set type `core.automobiles:CarInfo` to indicate that all properties of the property set type are considered as sensitive data:

```
{
  "Name": "core.automobiles:CarInfo",
  "DataCategory": "MasterData",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Master data for car information"
  }],
  "Annotations": [{
    "Name": "com.sap.appiot.security:spi",
    "PackageName": "com.sap.appiot.security"
  }]
}
```

```

    }],
    "Properties": [{
      "Name": "RegNum",
      "Descriptions": [{
        "LanguageCode": "en",
        "Description": "registration number"
      }],
      "Type": "String",
      "PropertyLength": "225"
      "UnitOfMeasure": "F"
    }]
  }
}

```

## Event Property Set Type Configuration (OData Service)

In the following example payload for creating an event property set type, the predefined annotation `com.sap.appiot.security:spi` is assigned to the property set type `core.automobiles:TemparatureEventPST` to indicate that all properties of the property set type are considered as sensitive data:

```

{
  "Name": "core.automobiles:TemparatureEventPST",
  "DataCategory": "EventData",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Properties in core.automobiles:TemparatureEventPST"
  }],
  "Annotations": [{
    "Name": "com.sap.appiot.security:spi"
  }],
  "Properties": [{
    "Name": "Temperature",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Temperature of the Engine"
    }],
    "Type": "Numeric",
    "PropertyLength": "",
    "QualityCode": "1",
    "UnitOfMeasure": "AA"
  },
  {
    "Name": "Pressure",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Pressure of the Engine"
    }],
    "Type": "Numeric",
    "PropertyLength": "",
    "QualityCode": "0",
    "UnitOfMeasure": "BA"
  }
]
}

```

## Related Information

[SAP Leonardo IoT Security Guide](#)

[Annotations \(OData Service\) \[page 501\]](#)

## 6.1 Delete User Data

With the various micro services provided by SAP Leonardo IoT, you can configure metadata objects such as package or thing type and create data for the metadata objects. These services when called persists the user data field `username`. If the user's consent to persist personal or private data is withdrawn, you must delete such data. You can use the service described in this topic to delete the user data `username`.

Deletion of user data is not a one-time activity. Though the user data is deleted, it is persisted again whenever the metadata objects created by the user is processed (created, modified, or deleted) by the API services. Hence, you must always ensure to delete the user data based on the user's consent.

### Request

**URI:** `http://<server address>[:<port number>][/path]/UserInfo`

**Example for a base URI in a cloud foundry environment:** `https://sap-iotaexplore.iot-sap.cfapps.eu10.hana.ondemand.com/apiot-mds/UserInfo`

**HTTP Method:** `DELETE`

**Permissions:** `<thing>.d`

### Request Parameters

None

### Request Example

```
/UserInfo
```

### Response

#### Response Status and Error Codes

Code	Description
200	User data deleted successfully
403	Forbidden

## 6.2 Export Rule Data

Lets you export all the data related to rules and rule contexts.

As a preparatory step for customer offboarding, SAP offers this service method that can be used to export all the data related to rules and rule contexts. With that, you have the opportunity to move to a platform other



than SAP Leonardo IoT without losing rule-related data. The method collects all rule-related data available in the tenant and exports it into a data stream that can be stored as a ZIP file.

SAP Leonardo IoT makes use of SAP Cloud Platform Business Rules as an underlying rule engine. This means that while you are creating and maintaining objects with the rule modeling tools of SAP Leonardo IoT, the system automatically reflects that in corresponding secondary objects handled by SAP Cloud Platform Business Rules. This is true, e.g., for rule conditions and event names. When using the rule export service, you can decide whether the system shall export these corresponding objects as well, depending on a parameter setting provided for that purpose.

### Note

When using this service, keep the following in mind:

- Exporting the corresponding rule objects along with the primary rule objects maintained in SAP Leonardo IoT is a time-consuming feature, which results in a response time slowdown of approximately 0.5 seconds per rule context. Therefore, if you want to export a high number of rule contexts, you should pay extra attention to the processing time.
- There is currently no import tool available that you could use to bring exported data back into the system.
- The internal format of the data files contained inside the ZIP file is subject to change. It is therefore **not** advisable to invest any efforts in developing any kind of tools that you might want to use for accessing the data.

## Request

**URI:** `v1/data-management/tenant`

**HTTP Method:** `GET`

**Permissions:**

`rules.r`

`voc.r`

### Request Header Fields

Name	Required	Description
<code>content-type</code>	Yes	<code>application/zip</code>

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
<code>addRuleCondition</code>	No	Boolean	Indicates whether secondary rule objects shall be exported along with the primary rule objects. Default value: <code>False</code>	Query string

## Request Example

v1/data-management/tenant?addRuleCondition=false

With this request, the system collects all the primary rule data that is present in the current tenant and returns a byte stream in its payload that represents the content of a ZIP file. The optional `addRuleCondition` parameter could as well have been omitted, because it defaults to `False` anyway.

## Response

After successful operation, the service returns a byte array that can be stored as a ZIP file in the file system for backup purposes. The internal structure of the resulting ZIP file is as follows: The ZIP file contains a flat list of data files. There is one data file for each rule as well as one for each rule context. If the method has been invoked with the `addRuleCondition` parameter set to `True`, a third set of data files is added to the ZIP file which contain the associations of rules with a rule context. The data files are named according to the following naming convention:

- Rule\_<Rule\_ID>
- Vocabulary\_<Rule\_Context\_ID>
- RuleDataForVocabulary\_<Rule\_Context\_ID>

In the files that belong to the third type, the following entries are logged for each rule:

- <Rule\_Name>\_<Creation\_Time>\_draft: Contains the rule data of the (inactive) draft version of the rule
- <Rule\_Name>\_<Creation\_Time>: Contains the rule data of the active version of the rule

## Response Status and Error Codes

Code	Reason
200	Rule data exported successfully.

## Response Example

Here is an example of a data file representing a rule:

```
{ "createdOn": "2019-05-14T07:22:56.663Z", "lastChangedOn": "2019-05-15T06:21:41.868Z",
  "id": "db7f2b98-dcb9-4c2e-baa4-ae6351721c90", "name": "VocabularyName_1557818569402",
  "eTag": "0f51bef3c244df0a483c91058c854958", "ermProjectId": "65f832f22d1c4e77870eab352506aa79",
  "texts": [ { "language": "en", "shortText": null, "description": "" } ],
  "thingModels": [ { "vocabularyId": "db7f2b98-dcb9-4c2e-baa4-ae6351721c90", "propertySet": "timeSeriesData",
    "thingType": "noah.sbrules02.rules:MultiVocThingType", "propertySetType": "noah.sbrules02.rules:MultiVocTimeSeries",
    "referencePropertySetType": "noah.sbrules02.rules:MultiVocReferencePst", "useThreshold": true, "modelType": "TimeSeriesData",
    "dataObjectName": "timeSeriesDOName" } ] }
```

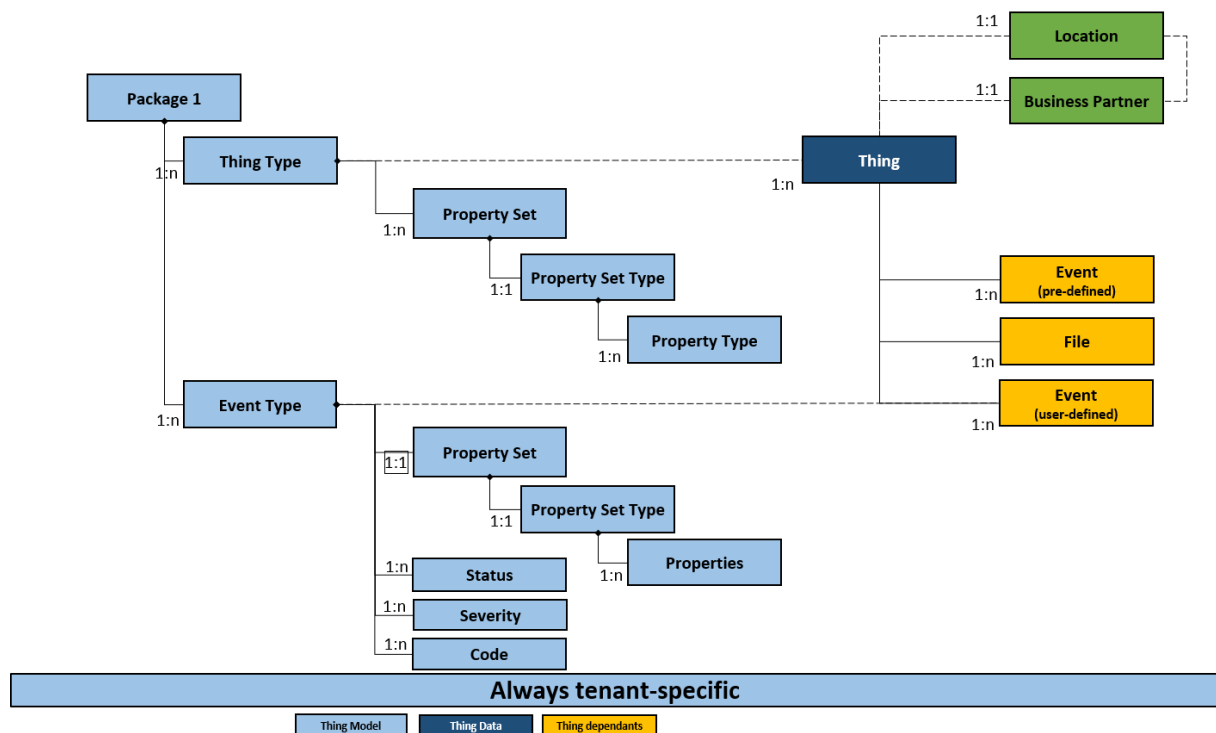
## Related Information

[Offboarding](#)

# 7 Configuration: Metadata Objects

A package is a container that contains metadata objects such as *ThingTypes*, *EventTypes*, *PropertySetTypes*, and *Property Types*. The Configuration service orchestrates the calls to other services such as thing metadata services and business partner metadata services to ensure a consistent system configuration.

The following diagram illustrates the various metadata objects:



## Property

The properties of a thing have a specific data type assigned. You can group the properties under a property set type. For example, the colour and model of the car, pressure and rotation speed of the wheel, or capacity and transmission of an engine are some of the property types.

The properties of the event property set type holds the measurement values based on which the events are triggered. The properties have a specific data type assigned. You can group the properties under a property set type. For example, the temperature of the engine and the message displayed when the temperature of engine exceeds the threshold value are some of the properties.

## Property Set Type

Property set type is a group of attributes that are used to describe things. Property set types enable groups of property types to be assigned to a thing. Property set types are assigned to thing types to describe a thing. Property set type can be assigned to one of the available data categories such as *MasterData*, *TimeSeriesData*, *DerivedData*, and *Parameters*. For example, **Car**, **Wheel**, and **Engine** are valid property set types. The colour

and model of the car, pressure and rotation speed of the wheel, and capacity and transmission of an engine are some of the property types that you can define for the corresponding property set type. Similarly, you can define property set types to be assigned to an event. These property set types belong to the data category [EventData](#) and are assigned to event types to describe an event. For example, **Exceeded Temperature** is a valid property set type defined for an event type. Temperature of the engine and the message displayed when the temperature exceeds are some of the properties that you can define for the corresponding property set type.

## Annotation

Annotation is an array of attributes used to qualify property types of a property set type. You can define annotations in a package and assign it only to property set types that belong to the same package. In addition to the annotations you define, you can also assign the predefined annotations offered by SAP.

## Thing Type

Thing type is used to group the property sets that describes a thing. For example, you can define a thing type Automobiles. You can create different things, for example, different automobiles identified by their license plates or vehicle identification numbers.

## Property Set

The assigned property set type for a thing type or an event type is called a property set. For example, property set type Wheel is assigned as property sets **FrontLeftWheel** and **FrontRightWheel** for a thing type.

## Thing

A thing is an instance of a thing type. Thing type and thing have a one-to-many relationship, that is, one thing type may be referenced by many things, while each thing is defined by exactly one thing type. For example, you can create a thing of a specific car model ABCXSeries of thing type Automobiles and specify values for the attributes defined in the Automobiles thing type.

## Event Type

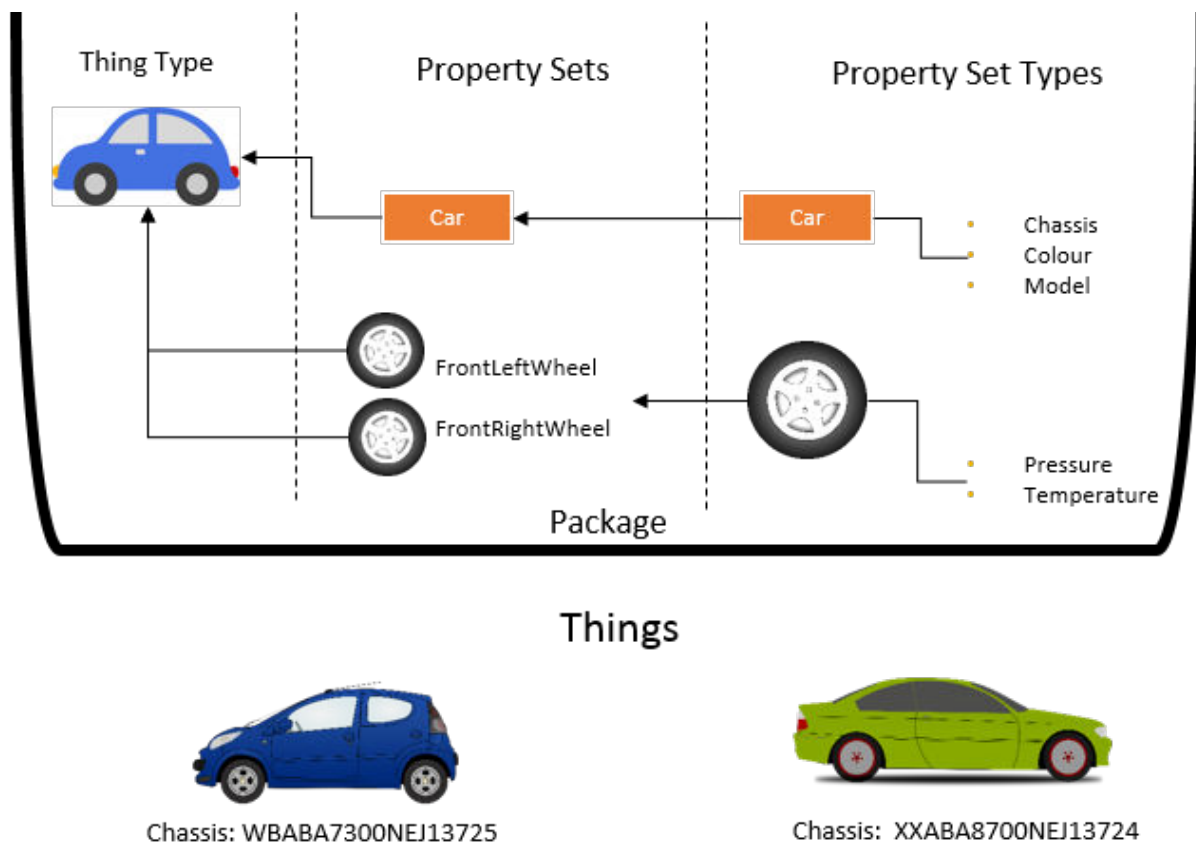
An event type is used to group the property sets that describe an event along with the different statuses and severities. You can define event types as part of the package configuration. You can create events of different event types to associate the change in data for a thing.

## Event

An event instance of an event type.

## Thing Model

The following diagram is an example thing model:



## Example

### Industrial Robot

Consider an industrial robot as a **Thing**. Industrial robots can be used in a production line of a car manufacturing plant. It can have different components such as a motor drive, hydraulic piston, and so on. These components carry sensors that provide relevant data about the robot at specific time intervals.

The Configuration service is used to model **Things**. These are the types of data that can be stored and read using the Thing service. Consider the motor drive component of the robot. The motor drive can have different signals from the sensors such as temperature, power consumption, motor status, and rotations per minute. The data that comes from these sensors can have different properties (such as temperature, power consumption) and units of measure (K, KWHr). This forms the basic atom of a **Thing** known as **PropertySetType**.

There can be many types of such industrial robots. A thing type contains an array of property set types. A robot of Type A can contain both property set types (for example, motor drive and piston) while another robot of Type B may contain only one of the property set types (for example, motor drive).

### Car Manufacturer

Consider a car manufacturer whose task is to monitor the performance of all manufactured cars out of the assigned production line. Cars of different models (for example, ABC2Series and ABCXSeries) produced by this manufacturer can have some attributes that are similar. In such case, each car model can be modeled as a thing type containing many property set types that in turn contain properties.

A few of the properties such as body style, class, wheel base can be master data that corresponds to non-time series data. Few of the properties such as pressure and temperature of a wheel can correspond to time series data (data that changes with time very frequently while the car is in operation).

## Related Information

[Reference Property Set Type \[page 331\]](#)

[Configuration: OData Service \[page 334\]](#)

## 7.1 Reference Property Set Type

You can define attributes such as unit of measure and threshold limits for a property using the configuration service.

In this way, all things and all data records of the property have the same unit of measure and threshold limit. In addition, SAP Leonardo IoT allows you to use the data category [ReferencePropertyData](#) to define reference properties with different predefined attribute types. In this way, all data records of a property for a thing instance can have different unit of measure and threshold limits at different points in time, as defined.

The following are the supported predefined attribute types:

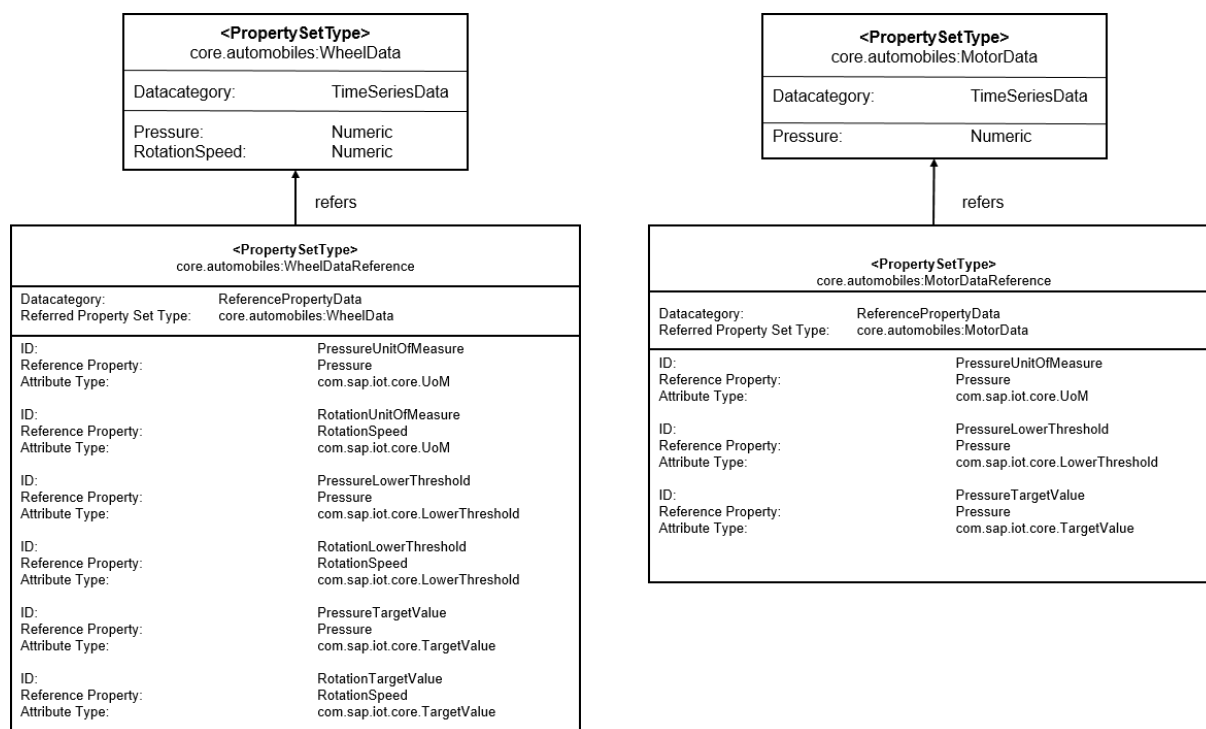
- `com.sap.iot.core.LowLowerThreshold`
- `com.sap.iot.core.LowerThreshold`
- `com.sap.iot.core.UpperThreshold`
- `com.sap.iot.core.UpperUpperThreshold`
- `com.sap.iot.core.UoM`
- `com.sap.iot.core.TargetValue`

### Note

- For a property type of attribute type `com.sap.iot.core.UoM` and `com.sap.iot.core.TargetValue`, the data must be of one of the valid predefined value data types.
- For a property type of other attribute types, `com.sap.iot.core.LowLowerThreshold`, `com.sap.iot.core.LowerThreshold`, `com.sap.iot.core.UpperThreshold`, and `com.sap.iot.core.UpperUpperThreshold`, the data must match data type [Numeric](#) or [NumericFlexible](#).
- You can define a property type of attribute type `com.sap.iot.core.TargetValue` to send a desired value to a device modeled using SAP Cloud Platform Internet of Things for the Cloud Foundry Environment. When you update or change the value of the property type, the device is updated with the new value once it is connected. To define this attribute, you must have already created [Credentials \[page 753\]](#).

For more information about sending data to a device, see [Send Data](#) in the product documentation of SAP Cloud Platform Internet of Things for the Cloud Foundry Environment.

The following diagram illustrates the reference property type:



- You can define a reference property set type for a property set type that is part of the current package, a dependent package, or a package to which you have access. In the above example, property set type **WheelData** and reference property set type **WheelDataReference** can be part of the same package, dependent packages, or a package to which you have access.
- You can refer to a specific property set type only once. In the above example, you cannot create another reference property set type using the property set type **WheelData**.
- You can only refer to a property set type of data category [TimeSeriesData](#). If the referred property set type is considered as personal information or sensitive personal information by assigning the predefined annotations (`com.sap.appiot.security:pii` and `com.sap.appiot.security:spi`), the reference property data such as threshold values, unit of measure, and target values are also treated as personal information or sensitive personal information. Hence, if data is created or modified for such reference properties, the system generates audit logs. However, you cannot directly assign these predefined annotations for the property set type of data category [ReferencePropertyData](#).
- You cannot reuse the reference property set type for another reference. In the above example, you cannot use the reference property set type **WheelDataReference** to create another reference property.
- You can define the property type as reference property in the reference property set type only if it belongs to the referred property set type. In the above example, reference property **Pressure** must belong to the referred property set type **WheelData**.
- You can refer to a specific attribute type for a reference property type only once. In the above example, you cannot define another instance of **Pressure** with the same attribute type `com.sap.iot.core.UoM`.
- You can only define the fields `id`, `referenceProperty`, and `attributeType` for a property type in the reference property set type.



- You cannot define a property type in the reference property set type without the fields *id*, *referenceProperty*, and *attributeType*.
- You cannot delete or update the reference property type if time series data exists for that property type.

## 8 Configuration: OData Service

Configuration-related services

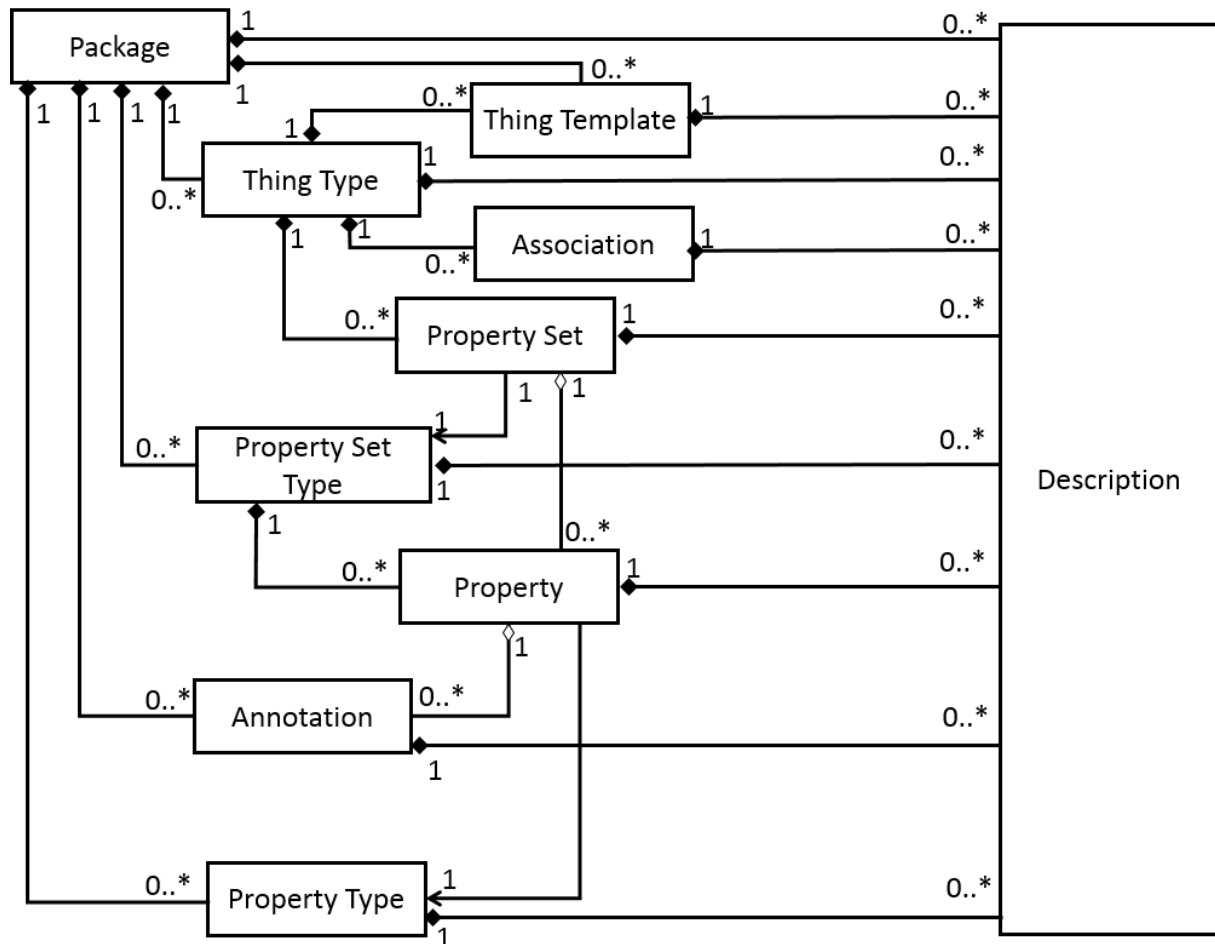
### Overview

You can use this service to configure individual objects such as package, thing type, property set type, and annotation in SAP Leonardo IoT platform. Hence, the service allows you to update the individual objects simultaneously within the same package. You need not maintain the version of all individual objects, instead the system generates and maintains version of the objects. Only the latest version of the objects is active for updates.

**OData Version: 2.0**

## Entity Data Model

The following diagram shows the EDM of the configuration package:



Here is a short overview of what the objects shown in the diagram represent and how they are related:

- All objects which, in their entirety, make up the thing model of SAP Leonardo IoT, are assigned to a **Package**. The package serves as a container for all objects that are set up to serve a particular business purpose, e.g., all kinds of objects that you need to model the digital twin of a car or a coffee machine. The package also serves as a virtual data space where all objects defined in the package are visible and can be accessed by each other.
- During thing modeling, you typically start by identifying the general aspects of a particular group of mostly identical things in the real world (coffee machine type "CM 3000", car type "Volkswagen Beetle"). These common aspects are reflected by a **Thing Type**.
- A thing type alone isn't much more than an empty bucket with a small set of attributes, such as a name and ID. To actually describe what is specific of a particular thing type, you set up so-called **Property Set Types** from which you then derive a **Property Set** that you can assign to one or many thing types. As the name suggests, a property set type is an object used for bundling a set of individual properties that belong together (e.g., the fill level and the temperature of a milk tank found in a coffee machine).
- Typically, each **Property** that is part of a property set type corresponds to a measured value that is recorded by a sensor mounted to a physical device. However, properties can also represent basic data of a

thing that is stable over time (e.g., name, color, serial number) or derived data that is calculated on the basis of measured data (e.g., average temperature within a given time frame).

- The technical details of each property (such as maximum string length or number of decimals) are stored in an object called **Property Type**. Each property refers to exactly one property type, while a package can contain any number of property types.
- Under real-world conditions, you will often feel the need to instantiate numerous variants of things that have most of their properties in common, but vary in certain aspects. E.g., as a producer of coffee machines, you may have different product lines with the same customer-facing features, but different tank volumes or compression unit pressure limits. For the purpose of reflecting such similar, but not identical products, you may take advantage of using **Thing Templates**. The thing template is derived from a thing type just like any other thing. However, you then define the common parts of things to be created in the thing template. Once done, you use the template for creating copies of the template as things. Each thing based on the template automatically inherits all settings that are present in the template, while the varying parts can be defined individually per thing.
- Sometimes, you may want to qualify certain properties found in a property set. You can accomplish this by associating a property with an **Annotation**. The most common use case for that may be to highlight certain properties as sensitive data or private data. Such qualification may be legally required by applicable law like the General Data Protection Regulation (GDPR), which is in effect in the European Union.
- Collecting sensor data for things is not a purpose in itself. Rather, you may feel the need to relate thing data with business-specific entities that you maintain in your ERP system, such as customer, supplier, sales order, storage location, and so on. To accomplish this, you can establish an **Association** between a thing type and a business object to which it is related.
- Each single real-world instance of the things mirrored by a thing type has its virtual equivalent in an object called **Thing** (e.g., the particular coffee machine of type CM 3000 with the unique serial number 987987945 that is installed in the coffee corner in the second floor of your company's main building).
- Each type of object can be semantically specified with a **Description** in natural language. Although it is good practice to provide such descriptions, it is technically not required and can therefore be omitted.

**Service Metadata URI for package:** `http://<server address>[:<port number>]/Package/v1/$metadata`

**Service Metadata URI for thing configuration:** `http://<server address>[:<port number>]/ThingConfiguration/v1/$metadata`

## Resources

Resource	Description	Path
<a href="#">Package Configuration [page 337]</a>	Package	<code>/v1/Packages</code>
<a href="#">Thing Type [page 368]</a>	Thing Type	<code>/v1/ThingTypes</code>
<a href="#">Thing Template [page 406]</a>	Thing Template	<code>/v1/ThingTemplates</code>
<a href="#">Property Set Type [page 434]</a>	Property Set Type	<code>/v1/PropertySetTypes</code>
<a href="#">Property Type [page 476]</a>	Property Type	<code>/v1/PropertyTypes</code>
<a href="#">Annotation [page 501]</a>	Annotation	<code>/v1/Annotations</code>

Resource	Description	Path
<a href="#">Event Configuration [page 512]</a>	Event	/v1/EventTypes

## Related Information

[Configuration: Metadata Objects \[page 328\]](#)

## 8.1 Package Configuration

Service to create, retrieve, and update the package objects

The configuration service allows you to create, retrieve, and update package objects. A package is a container that contains metadata objects such as [ThingTypes](#), [PropertySetTypes](#), and [Annotations](#). As the objects are managed individually, you can update the objects belonging to the same package simultaneously without locking the package. You can change the status of the package as [Freeze](#) and the service does not allow you to create new objects for a frozen package.

**Resource Path:** `http://<server address>[:<port number>][/path]/Package`

**Example for a base URI in a cloud foundry environment:** `https://config-package-sap.cfapps.eu10.hana.ondemand.com/Package`

### Operations

#### CRUD Operations

HTTP Method	Operation	URI	
<i>GET</i>	<a href="#">Read Metadata of Package [page 338]</a>	/v1/\$metadata	<pkg>.r
<i>POST</i>	<a href="#">Create a Package [page 341]</a>	/v1/Packages	<pkg>.c
<i>GET</i>	<a href="#">Read a Package [page 343]</a>	/v1/Packages ('<package name>')	<pkg>.r
<i>GET</i>	<a href="#">Read all Packages [page 345]</a>	/Package/v1/Packages	<pkg>.r
<i>PATCH</i>	<a href="#">Update a Package [page 349]</a>	/v1/Packages ('<package name>')	<pkg>.u
<i>DELETE</i>	<a href="#">Delete a Package [page 350]</a>	/v1/Packages ('<package name>')	<pkg>.d

#### i Note

<pkg> refers to the package configuration application name.

## Package Object Properties

Property	Data Type	Mandatory	Maximum Length	Description
Name	string	Yes	50	Unique name of the configuration package
Description	string	No	60	Description of the configuration package for each language with the corresponding ISO language code
Scope	string	No	10	Indicates the scope of the configuration package.  Permissible values are <i>tenant</i> or <i>private</i> .
Status	string	No	10	Indicates the status of the configuration package. Permissible values are <i>Active</i> or <i>Freeze</i> .

**i Note**  
Default value is *private*

**i Note**  
The value provided for this field in the request payload is ignored and the package is always created with the status *Active*.

## Package Description Properties

Property	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the object description
Description	String	255	Description of the object

## 8.1.1 Read Metadata of Package

With this method, you can retrieve the metadata of the *Package* entity.

### Request

URI: `/Package/v1/$metadata`

Operation Type: CRUD

HTTP Method: *GET*

Permissions: <pkg>.r

## Request Headers

None

## Request Parameters

None

## Request Example

```
/Package/v1/$metadata
```

## Response

### Response Status and Error Codes

Code	Description
200	Package metadata retrieved successfully

## Payload

Format: [XML](#)

Media types supported are [JSON](#) and [XML](#). The [XML](#) for this method has the following structure:

```
<EntityType Name="Package">
  <Documentation>
    <Summary>Details of package type entity</Summary>
  </Documentation>
  <Key>
    <PropertyRef Name="Name"></PropertyRef>
  </Key>
  <Property Name="Name" Type="Edm.String" Nullable="false"
MaxLength="50" sap:creatable="true" sap:updatable="false">
    <Documentation>
      <Summary>Name of package</Summary>
    </Documentation>
  </Property>
  <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="true" sap:updatable="true">
    <Documentation>
      <Summary>Description of package entity</Summary>
    </Documentation>
  </Property>
  <Property Name="Scope" Type="Edm.String" Nullable="true"
MaxLength="10" sap:creatable="true" sap:updatable="true">
    <Documentation>
      <Summary>Scope of package entity</Summary>
    </Documentation>
  </Property>
  <Property Name="Status" Type="Edm.String" Nullable="true"
MaxLength="10" sap:creatable="false" sap:updatable="true">
```

```

        <Documentation>
            <Summary>Status of the package</Summary>
        </Documentation>
    </Property>
    <NavigationProperty Name="Descriptions"
Relationship="com.sap.appcore.PackageToDescriptionAssociation"
FromRole="Package" ToRole="Description"></NavigationProperty>
</EntityType>
    <EntityType Name="Description">
        <Documentation>
            <Summary>Details of description type entity</Summary>
        </Documentation>
        <Key>
            <PropertyRef Name="LanguageCode"></PropertyRef>
        </Key>
        <Property Name="LanguageCode" Type="Edm.String" Nullable="false"
MaxLength="2" sap:creatable="true" sap:updatable="false">
            <Documentation>
                <Summary>Language code of description</Summary>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Description of package entity</Summary>
                <LongDescription>Long description of description entity</
LongDescription>
            </Documentation>
        </Property>
    </EntityType>
    <Association Name="PackageToDescriptionAssociation">
        <End Type="com.sap.appcore.Package" Multiplicity="1"
Role="Package"></End>
        <End Type="com.sap.appcore.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <EntityContainer Name="PackageContainer"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="Packages" EntityType="com.sap.appcore.Package">
            <Documentation>
                <Summary>Details of package set type entity</Summary>
            </Documentation>
        </EntitySet>
        <EntitySet Name="Descriptions"
EntityType="com.sap.appcore.Description">
            <Documentation>
                <Summary>Details of description set type</Summary>
            </Documentation>
        </EntitySet>
        <AssociationSet Name="PackageToDescriptionAssociation"
Association="com.sap.appcore.PackageToDescriptionAssociation">
            <End EntitySet="Packages" Role="Package"></End>
            <End EntitySet="Descriptions" Role="Description"></End>
        </AssociationSet>
    </EntityContainer>

```

## Related Information

[Package Configuration \[page 337\]](#)



## 8.1.2 Create a Package

With this method, you can create a package.

By default, the status of the package is set to [Active](#). The service ignores the status value specified in the request payload of the `POST` method. You can change the status of the package to [Freeze](#) while updating a package using the `PATCH` service. After update, the system saves the status of the package as [Frozen](#) in the database. You cannot edit a package with status [Frozen](#). The response header parameter `ETag` represents the entity value of the newly created package instance. This `ETag` value is a mandatory request header parameter used to update the latest version of a specific package using the `PATCH` service.

### Naming convention for package

- Must be a combination of alphanumeric characters and dots (dots to be used only to separate segments).
- Dots have to be followed by an alphanumeric character.
- Prefix the package name with the namespace of the tenant. For more information, see [Tenant \[page 282\]](#).
- Must be a unique name within the tenant.
- Do not start the name with a dot or digit.
- Any segment length has to be at least 2 characters.
- Second segment length has to be at least 3 characters.
- Maximum length is 50 characters.
- The name must contain a minimum of two segments.
- The name should be in lowercase.

#### Note

You can only create a package for a tenant, if a namespace is assigned for the tenant and the namespace is not null or empty.

### Scopes

Scope defined for a package determines whether the package and the metadata objects contained in the package are available for public access or not. The default scope for a package is always set to private. You can define the following scopes for a package:

- **private:** Metadata objects in such a package are visible only within the package. Hence, you cannot use the metadata objects of such a package in any other package.
- **tenant:** Metadata objects in such a package are visible only within a tenant where the package is created. Hence, you can use the metadata objects in such a package only in other packages belonging to the same tenant.

### Request

**URI:** `/Package/v1/Packages`

**Operation Type:** CRUD

**HTTP Method:** [POST](#)

**Permissions:** `<pkg>.c`

## Request Parameters

None

## Request Example

/Package/v1/Packages

## Payload

Format: *JSON*

```
{
  "Name": "core.automobiles",
  "Scope": "tenant",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Automobiles Package"
  }]
}
```

## Response

### Response Status and Error Codes

Code	Description
201	Package created successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

```
{
  "d": {
    "__metadata": {
      "id": "https://config-package-sap.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')",
      "uri": "https://config-package-sap.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')",
      "type": "com.sap.appcore.Package"
    },
    "Name": "core.automobiles",
    "Description": null,
    "Scope": "tenant",
    "Status": "Active",
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-package-sap.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')/Descriptions"
      }
    }
  }
}
```

```
}  
  }  
}
```

## 8.1.3 Read a Package

With this method, you can retrieve the details contained within a package with specific ID. The response returns the latest version of the package.

### Request

**URI:** `/Package/v1/Packages ('<Name>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<pkg>.r`

### Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	String	Unique name of the package

### Query String Parameters

Parameter	Required	Data Type	Description
\$expand	No	String	<p>Includes the related resources inline while retrieving the requested resources.</p> <p>The only association of a package that can be retrieved using \$expand is <a href="#">Description</a>.</p>

### Request Example

```
/Package/v1/Packages ('core.automobiles')
```

Retrieves details of the package with the ID `core.automobiles`

## Response

### Response Headers

Header	Description
ETag	Entity tag of the newly created package instance. The system generates a new value and replaces the old value every time a package is updated.

### Response Status and Error Codes

Code	Description
200	Package retrieved successfully
404	Specific package is not found

## Payload

**Format:** *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as depicted in the following examples:

#### Example 1

```
/Package/v1/Packages('core.automobiles')
```

Retrieves details of the package `core.automobiles`

```
{
  "d": {
    "__metadata": {
      "id": "https://config-package-sap-ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')",
      "uri": "https://config-package-sap-ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')",
      "type": "com.sap.appcore.Package"
    },
    "Name": "core.automobiles",
    "Description": "Automobiles Package",
    "Scope": "tenant",
    "Status": "Active",
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-package-sap-ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')/Descriptions"
      }
    }
  }
}
```

#### Example 2

```
/Package/v1/Packages('core.automobiles')?$expand=Descriptions
```

Retrieves details of the package `core.automobiles` with the list of descriptions in multiple languages, if provided.

```
{
  "d": {
    "__metadata": {
      "id": "https://config-package-sap-ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')",
      "uri": "https://config-package-sap-ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')",
      "type": "com.sap.appcore.Package"
    },
    "Name": "core.automobiles",
    "Description": "Automobiles Package",
    "Scope": "tenant",
    "Status": "Active",
    "Descriptions": {
      "results": [
        {
          "__metadata": {
            "id": "https://config-package-sap-ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Descriptions('en')",
            "uri": "https://config-package-sap-ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Descriptions('en')",
            "type": "com.sap.appcore.Description"
          },
          "LanguageCode": "en",
          "Description": "Automobiles Package"
        }
      ]
    }
  }
}
```

## 8.1.4 Read all Packages

With this method, you can retrieve all packages present in the database. The result set contains the latest version of all packages.

This method retrieves all packages to which the user has access authorization. It includes all packages with scope `tenant` and `private` from the same tenant to which the user has access.

In addition, you can retrieve a subset of all packages, according to the filter criteria provided. For a collective request for a set of packages, you have these options:

- You can restrict the set of packages to be retrieved by filter criteria based on the package name, description, scope, and status.
- You can define that the set of matching packages shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

URI: /Package/v1/Packages

Operation Type: [CRUD](#)

HTTP Method: [GET](#)

Permissions: <pkg>.r

### Request Parameters

None

### Query String Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set
\$skip	No	Integer	Number of the first n records to be excluded from the result set
\$orderby	No	Boolean	Field name to be used for sorting the result set in ascending order; valid fields are package name, description, scope, and status.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are package name, description, scope, and status.

Parameter	Required	Data Type	Description
\$inlinecount	No	String	<p>Total number of packages available; permissible values are <i>allpages</i> and <i>none</i>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"> <li>• If you use this with the \$top and \$skip parameters, the result contains the total count of hierarchies available.</li> <li>• If you use this with \$filter parameter, the result contains the total count of hierarchies based on the defined filter condition.</li> </ul>
\$filter	No	String	<p>Filter condition to be applied; the valid fields are package name, scope, status, and description. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the separators – <i>and</i>, <i>or</i>.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using \$filter. Valid fields are package name and description.</p>

## Request Example

```
/Package/v1/Packages
```

Retrieves all packages in the database

```
/Package/v1/Packages?$orderby=Name asc
```

Retrieves all packages and the retrieved packages are sorted by their name in ascending order.

```
/Package/v1/Packages?$select=Name, Scope
```

Retrieves all packages and retrieved result only contains the name and scope of the package

```
/Package/v1/Packages?$skip=10&$top=20
```

Retrieves 20 packages from the database where the first 10 packages are omitted. That is, the retrieved packages are those with sequence numbers 11 through 30.

## Response

### Response Status and Error Codes

Code	Description
200	Package retrieved successfully

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://config-package-sap-
ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')",
        "uri": "https://config-package-sap-
ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')",
        "type": "com.sap.appcore.Package"
      },
      "Name": "core.automobiles",
      "Description": "Automobiles Package",
      "Scope": "tenant",
      "Status": "Active",
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-package-sap-
ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.automobiles')/
Descriptions"
        }
      }
    }],
    {
      "__metadata": {
        "id": "https://config-package-sap-
ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.computers')",
        "uri": "https://config-package-sap-
ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.computers')",
        "type": "com.sap.appcore.Package"
      },
      "Name": "core.computers",
      "Description": "Computers Package",
      "Scope": "tenant",
      "Status": "Active",
      "Descriptions": {
```



```

        "__deferred": {
          "uri": "https://config-package-sap-
ci.cfapps.sap.hana.ondemand.com:443/Package/v1/Packages('core.computers')/
Descriptions"
        }
      }
    ]
  }
}

```

## 8.1.5 Update a Package

With this service, you can update an existing package.

The **ETag** value is mandatory to update a package. You must retrieve the **ETag** value of the package using the **GET** (`/Package/v1/Packages('<package name>')`) service. The **ETag** value is assigned to the request header parameter **If-Match** in the **PATCH** request. If you do not include the **ETag** value in the request header or if the **ETag** value is not correct, the server displays an error message.

Only the following actions are supported using this service:

- You can change the scope of the package from *private* to *tenant*.
- You can change the status of the package from *Active* to *Freeze*. After update, the system saves the status of the package as *Frozen*.
- You can change the description of the package.

After updating a package, the system generates a new **ETag** value for the package and replaces the old value with the new value.

### i Note

You cannot update a package with status *Frozen* using the **PATCH** service.

You cannot change the package name using **PATCH** request.

## Request

**URI:** `/Package/v1/Packages('<package name>')`

**Operation Type:** CRUD

**HTTP Method:** *PATCH*

**Permissions:** `<pkg>.u`

## Request Headers

Header	Required	Values
If-Match	Yes	Latest ETag value of a specific package that is updated using the PATCH request.

## Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	String	Unique name of the package

## Request Example

/Package/v1/Packages('core.automobiles')

```
{
  "Name": "core.automobiles",
  "Scope": "tenant",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Automobiles Package"
  }]
}
```

### Note

You cannot update the package name using the PATCH request. The package name specified in the request payload is ignored for update.

## Response

### Response Headers

Name	Description
Etag	Identifies the updated package

### Response Status and Error Codes

Code	Description
204	Package updated successfully.
400	Bad request.

## 8.1.6 Delete a Package

With this method, you can delete a package with a specific name.

The ETag value is mandatory to delete a package. You must retrieve the ETag value of the package using the GET (/Package/v1/Packages('<package name>')) service. The ETag value is assigned to the request

header parameter `If-Match` in the `DELETE` request. If you do not include the `ETag` value in the request header or if the `ETag` value is not correct, the server displays an error message.

### Note

You cannot delete a package under the following conditions:

- Metadata objects such as thing type, property set type, property type, annotation, event type, and event property set type are already created in the package being deleted.
- The package belongs to a different tenant.

## Request

**URI:** `/Package/v1/Packages('<package name>')`

**Operation Type:** CRUD

**HTTP Method:** `DELETE`

**Permissions:** `<pkg>.d`

### Request Headers

Header	Required	Values
<code>If-Match</code>	Yes	Latest <code>ETag</code> value of a specific package

### Request Parameters

Parameter	Required	Data Type	Description
<code>package name</code>	Yes	String	Unique name of the package

### Request Example

```
/Package/v1/Packages('core.automobiles')
```

## Response

### Response Status and Error Codes

Code	Description
204	Package deleted successfully

## 8.2 Thing Configuration

The thing configuration service uses the package to define the following metadata objects:

[Thing Type \[page 368\]](#)

[Thing Template \[page 406\]](#)

[Property Set Type \[page 434\]](#)

[Property Type \[page 476\]](#)

[Annotation \[page 501\]](#)

[Thing \[page 705\]](#)

### 8.2.1 Read Metadata of Thing Configuration

With this method, you can retrieve the metadata of thing configuration.

#### Request

**URI:** `/ThingConfiguration/v1/$metadata`

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** `<thingconf>.r`

#### Request Headers

None

#### Request Parameters

None

#### Request Example

`/ThingConfiguration/v1/$metadata`

## Response

### Response Status and Error Codes

Code	Description
200	Thing configuration metadata retrieved successfully.

### Payload

```
<EntityType Name="PropertySet">
  <Documentation>
    <Summary>Details of property set entity type</Summary>
  </Documentation>
  <Key>
    <PropertyRef Name="Name"></PropertyRef>
  </Key>
  <Property Name="Name" Type="Edm.String" Nullable="false"
MaxLength="30" sap:creatable="true" sap:updatable="false">
    <Documentation>
      <Summary>Name of entity</Summary>
    </Documentation>
  </Property>
  <Property Name="ThingType" Type="Edm.String" Nullable="false"
MaxLength="81" sap:creatable="false" sap:updatable="false">
    <Documentation>
      <Summary>Thing type of entity</Summary>
    </Documentation>
  </Property>
  <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="false">
    <Documentation>
      <Summary>Description of entity</Summary>
    </Documentation>
  </Property>
  <Property Name="PropertySetType" Type="Edm.String"
Nullable="false" MaxLength="81" sap:creatable="false" sap:updatable="false">
    <Documentation>
      <Summary>Details of property set type</Summary>
    </Documentation>
  </Property>
  <Property Name="DataCategory" Type="Edm.String" Nullable="false"
MaxLength="30" sap:creatable="false" sap:updatable="false">
    <Documentation>
      <Summary>Data category of entity</Summary>
    </Documentation>
  </Property>
  <Property Name="Operation" Type="Edm.String" Nullable="false"
MaxLength="10" sap:creatable="false" sap:updatable="false">
    <Documentation>
      <Summary>Operation to be performed for the object</
Summary>
    </Documentation>
  </Property>
  <NavigationProperty Name="ThingTypes"
Relationship="com.sap.apiot.PropertySetToThingTypeAssociation"
FromRole="PropertySet" ToRole="ThingType"></NavigationProperty>
  <NavigationProperty Name="PropertySetTypes"
Relationship="com.sap.apiot.PropertySetToPropertySetTypeAssociation"
FromRole="PropertySet" ToRole="PropertySetType"></NavigationProperty>
```

```

        <NavigationProperty Name="Properties"
Relationship="com.sap.apptot.PropertySetToPropertyAssociation"
FromRole="PropertySet" ToRole="Property"></NavigationProperty>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.PropertySetToDescriptionAssociation"
FromRole="PropertySet" ToRole="Description"></NavigationProperty>
    </EntityType>
    <EntityType Name="PropertySetType">
        <Documentation>
            <Summary>Details of property set type entity type</Summary>
        </Documentation>
        <Key>
            <PropertyRef Name="Name"></PropertyRef>
        </Key>
        <Property Name="Name" Type="Edm.String" Nullable="true"
MaxLength="81" sap:creatable="true" sap:updatable="false">
            <Documentation>
                <Summary>Name of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="PackageName" Type="Edm.String" Nullable="false"
MaxLength="50" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Package name of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="DataCategory" Type="Edm.String" Nullable="false"
MaxLength="30" sap:creatable="true" sap:updatable="false">
            <Documentation>
                <Summary>Data category of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Description of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="ReferredPropertySetType" Type="Edm.String"
Nullable="true" MaxLength="81" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Details of referred property set type entity
type</Summary>
            </Documentation>
        </Property>
        <Property Name="CopySource" Type="Edm.String" Nullable="true"
MaxLength="81" sap:creatable="true" sap:updatable="false">
            <Documentation>
                <Summary>Copy Source of Entity</Summary>
            </Documentation>
        </Property>
        <Property Name="NumberOfProperties" Type="Edm.Int16"
Nullable="true" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Property Type Count in a given Property Set
Type</Summary>
            </Documentation>
        </Property>
        <Property Name="IsMultiLingual" Type="Edm.Boolean"
Nullable="true" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Support for multiple language</Summary>
            </Documentation>
        </Property>
        <NavigationProperty Name="Properties"
Relationship="com.sap.apptot.PropertySetTypeToPropertyAssociation"
FromRole="PropertySetType" ToRole="Property"></NavigationProperty>

```

```

        <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.PropertyTypeToDescriptionAssociation"
FromRole="PropertySetType" ToRole="Description"></NavigationProperty>
        <NavigationProperty Name="Annotations"
Relationship="com.sap.apptot.PropertyTypeToAnnotationAssociation"
FromRole="PropertySetType" ToRole="Annotation"></NavigationProperty>
    </EntityType>
    <EntityType Name="Description">
        <Documentation>
            <Summary>Details of description type entity</Summary>
        </Documentation>
        <Key>
            <PropertyRef Name="LanguageCode"></PropertyRef>
        </Key>
        <Property Name="LanguageCode" Type="Edm.String" Nullable="false"
MaxLength="2" sap:creatable="true" sap:updatable="false">
            <Documentation>
                <Summary>Language code of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Description of entity</Summary>
                <LongDescription>Long description of description entity</
LongDescription>
            </Documentation>
        </Property>
    </EntityType>
    <EntityType Name="Property">
        <Documentation>
            <Summary>Details of property entity type</Summary>
        </Documentation>
        <Key>
            <PropertyRef Name="Name"></PropertyRef>
            <PropertyRef Name="PropertySetType"></PropertyRef>
        </Key>
        <Property Name="Name" Type="Edm.String" Nullable="false"
MaxLength="30" sap:creatable="true" sap:updatable="false">
            <Documentation>
                <Summary>Name of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="PropertySetType" Type="Edm.String"
Nullable="true" MaxLength="81" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Details of property set type</Summary>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Description of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="Type" Type="Edm.String" Nullable="true"
MaxLength="30" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Type of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="PropertyLength" Type="Edm.String"
Nullable="true" MaxLength="30" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Length of entity</Summary>
            </Documentation>
        </Property>

```

```

        <Property Name="QualityCode" Type="Edm.String" Nullable="true"
MaxLength="1" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Quality code of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="UnitOfMeasure" Type="Edm.String" Nullable="true"
MaxLength="5" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Unit of measure of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="ReferenceProperty" Type="Edm.String"
Nullable="true" MaxLength="115" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Reference property of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="AttributeType" Type="Edm.String" Nullable="true"
MaxLength="81" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Attribute of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="Operation" Type="Edm.String" Nullable="true"
MaxLength="10" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Operation to be performed for the object</
Summary>
            </Documentation>
        </Property>
        <Property Name="PropertyType" Type="Edm.String" Nullable="true"
MaxLength="81" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Details of Property Type Entity</Summary>
            </Documentation>
        </Property>
        <Property Name="IsMultiLingual" Type="Edm.Boolean"
Nullable="true" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Support for multiple language</Summary>
            </Documentation>
        </Property>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.PropertyToDescriptionAssociation"
FromRole="Property" ToRole="Description"></NavigationProperty>
        <NavigationProperty Name="Annotations"
Relationship="com.sap.apptot.PropertyToAnnotationAssociation"
FromRole="Property" ToRole="Annotation"></NavigationProperty>
        <NavigationProperty Name="ReferenceProperties"
Relationship="com.sap.apptot.PropertyToPropertyAssociation" FromRole="Property"
ToRole="ReferenceProperty"></NavigationProperty>
    </EntityType>
    <EntityType Name="Annotation">
        <Documentation>
            <Summary>Details of annotation entity</Summary>
        </Documentation>
        <Key>
            <PropertyRef Name="Name"></PropertyRef>
        </Key>
        <Property Name="Name" Type="Edm.String" Nullable="false"
MaxLength="81" sap:creatable="true" sap:updatable="false">
            <Documentation>
                <Summary>Name of entity</Summary>
            </Documentation>
        </Property>
        <Property Name="PackageName" Type="Edm.String" Nullable="true"
MaxLength="50" sap:creatable="false" sap:updatable="false">

```



```

        <Documentation>
            <Summary>Package name of entity</Summary>
        </Documentation>
    </Property>
    <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Description of entity</Summary>
        </Documentation>
    </Property>
    <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.AnnotationToDescriptionAssociation"
FromRole="Annotation" ToRole="Description"></NavigationProperty>
</EntityType>
<EntityType Name="ObjectAssociation">
    <Documentation>
        <Summary>Object Association Entity Type Summary</Summary>
    </Documentation>
    <Key>
        <PropertyRef Name="PrimaryEntity"></PropertyRef>
        <PropertyRef Name="AssociatedEntity"></PropertyRef>
        <PropertyRef Name="Relationship"></PropertyRef>
    </Key>
    <Property Name="AssociationEntityType" Type="Edm.String"
Nullable="false" MaxLength="50" sap:creatable="true" sap:updatable="false">
        <Documentation>
            <Summary>Object Association Entity Type Summary</Summary>
        </Documentation>
    </Property>
    <Property Name="PrimaryEntity" Type="Edm.String"
Nullable="false" MaxLength="81" sap:creatable="true" sap:updatable="false">
        <Documentation>
            <Summary>Object Association Primary Entity Summary</
Summary>
        </Documentation>
    </Property>
    <Property Name="PrimaryTenant" Type="Edm.String"
Nullable="false" MaxLength="32" sap:creatable="true" sap:updatable="false">
        <Documentation>
            <Summary>Object Association Primary Tenant Summary</
Summary>
        </Documentation>
    </Property>
    <Property Name="AssociatedEntity" Type="Edm.String"
Nullable="false" MaxLength="81" sap:creatable="true" sap:updatable="false">
        <Documentation>
            <Summary>Object Association Associated Entity Summary</
Summary>
        </Documentation>
    </Property>
    <Property Name="AssociationType" Type="Edm.String"
Nullable="false" MaxLength="50" sap:creatable="true" sap:updatable="false">
        <Documentation>
            <Summary>Object Association Type Summary</Summary>
        </Documentation>
    </Property>
    <Property Name="Relationship" Type="Edm.String" Nullable="false"
MaxLength="50" sap:creatable="true" sap:updatable="false">
        <Documentation>
            <Summary>Object Association Relationship Summary</
Summary>
        </Documentation>
    </Property>
    <Property Name="AssociationExpiryTime" Type="Edm.String"
Nullable="true" MaxLength="50" sap:creatable="true" sap:updatable="false">
        <Documentation>
            <Summary>Object Association Expire Time Summary</Summary>
        </Documentation>
    </Property>

```

```

        </Property>
        <Property Name="AssociationCreationTime" Type="Edm.String"
Nullable="true" MaxLength="50" sap:creatable="true" sap:updatable="false">
        <Documentation>
        <Summary>Object Association Creation Time Summary</
Summary>
        </Documentation>
        </Property>
        <Property Name="AssociatedTenant" Type="Edm.String"
Nullable="false" MaxLength="32" sap:creatable="true" sap:updatable="false">
        <Documentation>
        <Summary>Object Association Associated Tenant Summary.</
Summary>
        </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="true" sap:updatable="true">
        <Documentation>
        <Summary>Description of entity</Summary>
        </Documentation>
        </Property>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.ObjectAssociationToDescriptionAssociation"
FromRole="ObjectAssociation" ToRole="Description"></NavigationProperty>
    </EntityType>
    <EntityType Name="Value">
        <Documentation>
        <Summary>Values supported in Entity</Summary>
        </Documentation>
        <Key>
        <PropertyRef Name="Value"></PropertyRef>
        </Key>
        <Property Name="Value" Type="Edm.String" Nullable="false"
MaxLength="254" sap:creatable="true" sap:updatable="true">
        <Documentation>
        <Summary>Values supported in Entity</Summary>
        </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="false">
        <Documentation>
        <Summary>Description of Values</Summary>
        </Documentation>
        </Property>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.ValueToDescriptionAssociation" FromRole="Value"
ToRole="Description"></NavigationProperty>
    </EntityType>
    <EntityType Name="Package">
        <Documentation>
        <Summary>Details of package entity type</Summary>
        </Documentation>
        <Key>
        <PropertyRef Name="Name"></PropertyRef>
        </Key>
        <Property Name="Name" Type="Edm.String" Nullable="false"
MaxLength="50" sap:creatable="false" sap:updatable="false">
        <Documentation>
        <Summary>Package name of entity</Summary>
        </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="false">
        <Documentation>
        <Summary>Description of entity</Summary>
        </Documentation>
        </Property>

```

```

        <Property Name="Scope" Type="Edm.String" Nullable="true"
MaxLength="10" sap:creatable="false" sap:updatable="false">
        <Documentation>
        <Summary>Scope of package</Summary>
        </Documentation>
    </Property>
    <Property Name="Status" Type="Edm.String" Nullable="true"
MaxLength="10" sap:creatable="false" sap:updatable="false">
        <Documentation>
        <Summary>Status of package</Summary>
        </Documentation>
    </Property>
    <NavigationProperty Name="ThingTypes"
Relationship="com.sap.apptot.PackageToThingTypeAssociation" FromRole="Package"
ToRole="ThingType"></NavigationProperty>
    <NavigationProperty Name="ThingTemplates"
Relationship="com.sap.apptot.PackageToThingTemplateAssociation"
FromRole="Package" ToRole="ThingTemplate"></NavigationProperty>
    <NavigationProperty Name="PropertySetTypes"
Relationship="com.sap.apptot.PackageToPropertySetTypeAssociation"
FromRole="Package" ToRole="PropertySetType"></NavigationProperty>
    <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.PackageToDescriptionAssociation" FromRole="Package"
ToRole="Description"></NavigationProperty>
    <NavigationProperty Name="Annotations"
Relationship="com.sap.apptot.PackageToAnnotationAssociation" FromRole="Package"
ToRole="Annotation"></NavigationProperty>
    <NavigationProperty Name="PropertyTypes"
Relationship="com.sap.apptot.PackageToPropertyTypeAssociation"
FromRole="Package" ToRole="PropertyType"></NavigationProperty>
    </EntityType>
    <EntityType Name="DYN_ENT_TemplateProperties">
        <Documentation>
        <Summary>Details of TemplateProperties Entity Set</Summary>
        </Documentation>
        <Property Name="PstName" Type="Edm.String" Nullable="true"
sap:creatable="true" sap:updatable="false"></Property>
        <Property Name="PropertyName" Type="Edm.String" Nullable="true"
sap:creatable="true" sap:updatable="false"></Property>
        <Property Name="PropertyValue" Type="Edm.String" Nullable="true"
sap:creatable="false" sap:updatable="false"></Property>
        <NavigationProperty Name="DYN_ENT_NamedPST"
Relationship="com.sap.apptot.TemplatePropertiesToNamedPST"
FromRole="DYN_ENT_TemplateProperties" ToRole="DYN_ENT_NamedPST"></
NavigationProperty>
    </EntityType>
    <EntityType Name="ThingTemplate">
        <Documentation>
        <Summary>Details of thing template</Summary>
        </Documentation>
        <Key>
        <PropertyRef Name="Name"></PropertyRef>
        </Key>
        <Property Name="Name" Type="Edm.String" Nullable="false"
MaxLength="81" sap:creatable="true" sap:updatable="false">
        <Documentation>
        <Summary>Name of the Entity</Summary>
        </Documentation>
    </Property>
        <Property Name="ThingType" Type="Edm.String" Nullable="false"
MaxLength="81" sap:creatable="true" sap:updatable="false">
        <Documentation>
        <Summary>Details of thing name</Summary>
        </Documentation>
    </Property>
        <Property Name="PackageName" Type="Edm.String" Nullable="false"
MaxLength="50" sap:creatable="false" sap:updatable="false">
        <Documentation>

```

```

        <Summary>Details of Associated Thing Template Package</
Summary>
        </Documentation>
    </Property>
    <Property Name="HasFile" Type="Edm.Boolean" Nullable="true"
sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>If the Template has file</Summary>
        </Documentation>
    </Property>
    <Property Name="FileData" Type="Edm.String" Nullable="true"
sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Details of Associated Thing Template Package</
Summary>
        </Documentation>
    </Property>
    <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Details of thing template description</Summary>
        </Documentation>
    </Property>
    <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.ThingTemplateToDescriptionAssociation"
FromRole="ThingTemplate" ToRole="Description"></NavigationProperty>
    <NavigationProperty Name="DYN_ENT_TemplateProperties"
Relationship="com.sap.apptot.ThingTemplateToTemplateProperties"
FromRole="ThingTemplate" ToRole="DYN_ENT_TemplateProperties"></
NavigationProperty>
</EntityType>
<EntityType Name="DYN_ENT_NamedPST">
    <Documentation>
        <Summary>Dynamic Property</Summary>
    </Documentation>
    <Property Name="DYN_ENT_PstName" Type="Edm.String"
Nullable="false" MaxLength="81" sap:creatable="true" sap:updatable="false"></
Property>
</EntityType>
<EntityType Name="PropertyType">
    <Documentation>
        <Summary>Details of Property Type Entity</Summary>
    </Documentation>
    <Key>
        <PropertyRef Name="Name"></PropertyRef>
    </Key>
    <Property Name="Name" Type="Edm.String" Nullable="false"
MaxLength="81" sap:creatable="true" sap:updatable="false">
        <Documentation>
            <Summary>Name of entity</Summary>
        </Documentation>
    </Property>
    <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Description of entity</Summary>
        </Documentation>
    </Property>
    <Property Name="Type" Type="Edm.String" Nullable="false"
MaxLength="30" sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Type of entity</Summary>
        </Documentation>
    </Property>
    <Property Name="PropertyLength" Type="Edm.String"
Nullable="true" MaxLength="30" sap:creatable="true" sap:updatable="true">
        <Documentation>
            <Summary>Length of entity</Summary>

```

```

        </Documentation>
    </Property>
    <Property Name="QualityCode" Type="Edm.String" Nullable="true"
MaxLength="1" sap:creatable="true" sap:updatable="true">
        <Documentation>
            <Summary>Quality code of entity</Summary>
        </Documentation>
    </Property>
    <Property Name="UnitOfMeasure" Type="Edm.String" Nullable="true"
MaxLength="5" sap:creatable="true" sap:updatable="true">
        <Documentation>
            <Summary>Unit of measure of entity</Summary>
        </Documentation>
    </Property>
    <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.PropertyTypeToDescriptionAssociation"
FromRole="PropertyType" ToRole="Description"></NavigationProperty>
    <NavigationProperty Name="Values"
Relationship="com.sap.apptot.PropertyTypeToValueAssociation"
FromRole="PropertyType" ToRole="Value"></NavigationProperty>
</EntityType>
<EntityType Name="ThingType">
    <Documentation>
        <Summary>Details of thing type</Summary>
    </Documentation>
    <Key>
        <PropertyRef Name="Name"></PropertyRef>
    </Key>
    <Property Name="Name" Type="Edm.String" Nullable="false"
MaxLength="81" sap:creatable="true" sap:updatable="false">
        <Documentation>
            <Summary>Details of thing name</Summary>
        </Documentation>
    </Property>
    <Property Name="PackageName" Type="Edm.String" Nullable="false"
MaxLength="50" sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Details of thing package name</Summary>
        </Documentation>
    </Property>
    <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Details of thing type description</Summary>
        </Documentation>
    </Property>
    <Property Name="ExtendedThingType" Type="Edm.String"
Nullable="true" MaxLength="81" sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Thing type which is extended by current thing
type</Summary>
        </Documentation>
    </Property>
    <Property Name="CopySource" Type="Edm.String" Nullable="true"
MaxLength="81" sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Thing Type which is copied by current Thing
Type</Summary>
        </Documentation>
    </Property>
    <NavigationProperty Name="PropertySets"
Relationship="com.sap.apptot.ThingTypeToPropertySetAssociation"
FromRole="ThingType" ToRole="PropertySet"></NavigationProperty>
    <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.ThingTypeToDescriptionAssociation"
FromRole="ThingType" ToRole="Description"></NavigationProperty>

```

```

        <NavigationProperty Name="ThingTemplates"
Relationship="com.sap.appiot.ThingTypeToThingTemplateAssociation"
FromRole="ThingType" ToRole="ThingTemplate"></NavigationProperty>
        <NavigationProperty Name="ObjectAssociations"
Relationship="com.sap.appiot.ThingTypeToObjectAssociation" FromRole="ThingType"
ToRole="ObjectAssociation"></NavigationProperty>
    </EntityType>
    <Association Name="ThingTypeToPropertySetAssociation">
        <End Type="com.sap.appiot.ThingType" Multiplicity="1"
Role="ThingType"></End>
        <End Type="com.sap.appiot.PropertySet" Multiplicity="*"
Role="PropertySet"></End>
    </Association>
    <Association Name="PropertySetTypeToDescriptionAssociation">
        <End Type="com.sap.appiot.PropertySetType" Multiplicity="1"
Role="PropertySetType"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <Association Name="PropertyToDescriptionAssociation">
        <End Type="com.sap.appiot.Property" Multiplicity="1"
Role="Property"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <Association Name="ObjectAssociationToDescriptionAssociation">
        <End Type="com.sap.appiot.ObjectAssociation" Multiplicity="1"
Role="ObjectAssociation"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <Association Name="PackageToPropertySetTypeAssociation">
        <End Type="com.sap.appiot.Package" Multiplicity="1"
Role="Package"></End>
        <End Type="com.sap.appiot.PropertySetType" Multiplicity="*"
Role="PropertySetType"></End>
    </Association>
    <Association Name="PropertyTypeToValueAssociation">
        <End Type="com.sap.appiot.PropertyType" Multiplicity="1"
Role="PropertyType"></End>
        <End Type="com.sap.appiot.Value" Multiplicity="*" Role="Value"></
End>
    </Association>
    <Association Name="PropertySetToPropertySetTypeAssociation">
        <End Type="com.sap.appiot.PropertySet" Multiplicity="1"
Role="PropertySet"></End>
        <End Type="com.sap.appiot.PropertySetType" Multiplicity="1"
Role="PropertySetType"></End>
    </Association>
    <Association Name="PropertyToPropertyAssociation">
        <End Type="com.sap.appiot.Property" Multiplicity="1"
Role="Property"></End>
        <End Type="com.sap.appiot.Property" Multiplicity="*"
Role="ReferenceProperty"></End>
    </Association>
    <Association Name="PackageToAnnotationAssociation">
        <End Type="com.sap.appiot.Package" Multiplicity="1"
Role="Package"></End>
        <End Type="com.sap.appiot.Annotation" Multiplicity="*"
Role="Annotation"></End>
    </Association>
    <Association Name="PropertySetToPropertyAssociation">
        <End Type="com.sap.appiot.PropertySet" Multiplicity="1"
Role="PropertySet"></End>
        <End Type="com.sap.appiot.Property" Multiplicity="*"
Role="Property"></End>
    </Association>
    <Association Name="ThingTypeToThingTemplateAssociation">

```

```

        <End Type="com.sap.appiot.ThingType" Multiplicity="1"
Role="ThingType"></End>
        <End Type="com.sap.appiot.ThingTemplate" Multiplicity="*"
Role="ThingTemplate"></End>
    </Association>
    <Association Name="PropertySetToThingTypeAssociation">
        <End Type="com.sap.appiot.PropertySet" Multiplicity="*"
Role="PropertySet"></End>
        <End Type="com.sap.appiot.ThingType" Multiplicity="1"
Role="ThingType"></End>
    </Association>
    <Association Name="PropertySetToDescriptionAssociation">
        <End Type="com.sap.appiot.PropertySet" Multiplicity="1"
Role="PropertySet"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <Association Name="PropertySetTypeToAnnotationAssociation">
        <End Type="com.sap.appiot.PropertySetType" Multiplicity="1"
Role="PropertySetType"></End>
        <End Type="com.sap.appiot.Annotation" Multiplicity="*"
Role="Annotation"></End>
    </Association>
    <Association Name="PropertySetTypeToPropertyAssociation">
        <End Type="com.sap.appiot.PropertySetType" Multiplicity="1"
Role="PropertySetType"></End>
        <End Type="com.sap.appiot.Property" Multiplicity="*"
Role="Property"></End>
    </Association>
    <Association Name="PropertyTypeToDescriptionAssociation">
        <End Type="com.sap.appiot.PropertyType" Multiplicity="1"
Role="PropertyType"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <Association Name="PackageToDescriptionAssociation">
        <End Type="com.sap.appiot.Package" Multiplicity="1"
Role="Package"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <Association Name="ThingTypeToObjectAssociation">
        <End Type="com.sap.appiot.ThingType" Multiplicity="1"
Role="ThingType"></End>
        <End Type="com.sap.appiot.ObjectAssociation" Multiplicity="*"
Role="ObjectAssociation"></End>
    </Association>
    <Association Name="AnnotationToDescriptionAssociation">
        <End Type="com.sap.appiot.Annotation" Multiplicity="1"
Role="Annotation"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <Association Name="PackageToPropertyTypeAssociation">
        <End Type="com.sap.appiot.Package" Multiplicity="1"
Role="Package"></End>
        <End Type="com.sap.appiot.PropertyType" Multiplicity="*"
Role="PropertyType"></End>
    </Association>
    <Association Name="TemplatePropertiesToNamedPST">
        <End Type="com.sap.appiot.DYN_ENT_TemplateProperties"
Multiplicity="1" Role="DYN_ENT_TemplateProperties"></End>
        <End Type="com.sap.appiot.DYN_ENT_NamedPST" Multiplicity="*"
Role="DYN_ENT_NamedPST"></End>
    </Association>
    <Association Name="ThingTypeToDescriptionAssociation">
        <End Type="com.sap.appiot.ThingType" Multiplicity="1"
Role="ThingType"></End>

```

```

        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <Association Name="PackageToThingTypeAssociation">
        <End Type="com.sap.appiot.Package" Multiplicity="1"
Role="Package"></End>
        <End Type="com.sap.appiot.ThingType" Multiplicity="*"
Role="ThingType"></End>
    </Association>
    <Association Name="PropertyToAnnotationAssociation">
        <End Type="com.sap.appiot.Property" Multiplicity="1"
Role="Property"></End>
        <End Type="com.sap.appiot.Annotation" Multiplicity="*"
Role="Annotation"></End>
    </Association>
    <Association Name="ValueToDescriptionAssociation">
        <End Type="com.sap.appiot.Value" Multiplicity="1" Role="Value"></
End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <Association Name="PackageToThingTemplateAssociation">
        <End Type="com.sap.appiot.Package" Multiplicity="1"
Role="Package"></End>
        <End Type="com.sap.appiot.ThingTemplate" Multiplicity="*"
Role="ThingTemplate"></End>
    </Association>
    <Association Name="ThingTemplateToDescriptionAssociation">
        <End Type="com.sap.appiot.ThingTemplate" Multiplicity="1"
Role="ThingTemplate"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <Association Name="ThingTemplateToTemplateProperties">
        <End Type="com.sap.appiot.ThingTemplate" Multiplicity="1"
Role="ThingTemplate"></End>
        <End Type="com.sap.appiot.DYN_ENT_TemplateProperties"
Multiplicity="*" Role="DYN_ENT_TemplateProperties"></End>
    </Association>
    <EntityContainer Name="ThingConfigurationContainer"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="PropertySets"
EntityType="com.sap.appiot.PropertySet">
            <Documentation>
                <Summary>Details of property set entity set type</
Summary>
            </Documentation>
        </EntitySet>
        <EntitySet Name="PropertySetTypes"
EntityType="com.sap.appiot.PropertySetType">
            <Documentation>
                <Summary>Details of property set type entity set</
Summary>
            </Documentation>
        </EntitySet>
        <EntitySet Name="Descriptions"
EntityType="com.sap.appiot.Description">
            <Documentation>
                <Summary>Details of description entity set type</Summary>
            </Documentation>
        </EntitySet>
        <EntitySet Name="Properties"
EntityType="com.sap.appiot.Property">
            <Documentation>
                <Summary>of property entity set type</Summary>
            </Documentation>
        </EntitySet>

```



```

        <EntitySet Name="Annotations"
EntityType="com.sap.appiot.Annotation">
        <Documentation>
            <Summary>Details of annotation entity set</Summary>
        </Documentation>
    </EntitySet>
    <EntitySet Name="ObjectAssociations"
EntityType="com.sap.appiot.ObjectAssociation">
        <Documentation>
            <Summary>ObjectAssociation Entity Set Summary</Summary>
        </Documentation>
    </EntitySet>
    <EntitySet Name="Values" EntityType="com.sap.appiot.Value">
        <Documentation>
            <Summary>Values supported in Entity</Summary>
        </Documentation>
    </EntitySet>
    <EntitySet Name="Packages" EntityType="com.sap.appiot.Package">
        <Documentation>
            <Summary>Details of thing type entity set</Summary>
        </Documentation>
    </EntitySet>
    <EntitySet Name="DYN_ENT_TemplateProperties"
EntityType="com.sap.appiot.DYN_ENT_TemplateProperties">
        <Documentation>
            <Summary>Details of TemplateProperties Entity Set</
Summary>
        </Documentation>
    </EntitySet>
    <EntitySet Name="ThingTemplates"
EntityType="com.sap.appiot.ThingTemplate">
        <Documentation>
            <Summary>Details of Thing Template Entity Set</Summary>
        </Documentation>
    </EntitySet>
    <EntitySet Name="DYN_ENT_NamedPST"
EntityType="com.sap.appiot.DYN_ENT_NamedPST">
        <Documentation>
            <Summary>Details of NamedPST EntitySet</Summary>
        </Documentation>
    </EntitySet>
    <EntitySet Name="PropertyTypes"
EntityType="com.sap.appiot.PropertyType">
        <Documentation>
            <Summary>Details of Property Type Entity</Summary>
        </Documentation>
    </EntitySet>
    <EntitySet Name="ThingTypes"
EntityType="com.sap.appiot.ThingType">
        <Documentation>
            <Summary>Details of thing type entity set</Summary>
        </Documentation>
    </EntitySet>
    <AssociationSet Name="ThingTypeToPropertySetAssociation"
Association="com.sap.appiot.ThingTypeToPropertySetAssociation">
        <End EntitySet="ThingTypes" Role="ThingType"></End>
        <End EntitySet="PropertySets" Role="PropertySet"></End>
    </AssociationSet>
    <AssociationSet Name="PropertySetTypeToDescriptionAssociation"
Association="com.sap.appiot.PropertySetTypeToDescriptionAssociation">
        <End EntitySet="PropertySetTypes" Role="PropertySetType"></
End>
        <End EntitySet="Descriptions" Role="Description"></End>
    </AssociationSet>
    <AssociationSet Name="PropertyToDescriptionAssociation"
Association="com.sap.appiot.PropertyToDescriptionAssociation">
        <End EntitySet="Properties" Role="Property"></End>
        <End EntitySet="Descriptions" Role="Description"></End>

```

```

        </AssociationSet>
        <AssociationSet Name="ObjectAssociationToDescriptionAssociation"
Association="com.sap.appiot.ObjectAssociationToDescriptionAssociation">
            <End EntitySet="ObjectAssociations"
Role="ObjectAssociation"></End>
            <End EntitySet="Descriptions" Role="Description"></End>
        </AssociationSet>
        <AssociationSet Name="PackageToPropertySetTypeAssociation"
Association="com.sap.appiot.PackageToPropertySetTypeAssociation">
            <End EntitySet="Packages" Role="Package"></End>
            <End EntitySet="PropertySetTypes" Role="PropertySetType"></
End>
        </AssociationSet>
        <AssociationSet Name="PropertyTypeToValueAssociation"
Association="com.sap.appiot.PropertyTypeToValueAssociation">
            <End EntitySet="PropertyTypes" Role="PropertyType"></End>
            <End EntitySet="Values" Role="Value"></End>
        </AssociationSet>
        <AssociationSet Name="PropertySetToPropertySetTypeAssociation"
Association="com.sap.appiot.PropertySetToPropertySetTypeAssociation">
            <End EntitySet="PropertySets" Role="PropertySet"></End>
            <End EntitySet="PropertySetTypes" Role="PropertySetType"></
End>
        </AssociationSet>
        <AssociationSet Name="PropertyToPropertyAssociation"
Association="com.sap.appiot.PropertyToPropertyAssociation">
            <End EntitySet="Properties" Role="Property"></End>
            <End EntitySet="Properties" Role="ReferenceProperty"></End>
        </AssociationSet>
        <AssociationSet Name="PackageToAnnotationAssociation"
Association="com.sap.appiot.PackageToAnnotationAssociation">
            <End EntitySet="Packages" Role="Package"></End>
            <End EntitySet="Annotations" Role="Annotation"></End>
        </AssociationSet>
        <AssociationSet Name="PropertySetToPropertyAssociation"
Association="com.sap.appiot.PropertySetToPropertyAssociation">
            <End EntitySet="PropertySets" Role="PropertySet"></End>
            <End EntitySet="Properties" Role="Property"></End>
        </AssociationSet>
        <AssociationSet Name="ThingTypeToThingTemplateAssociation"
Association="com.sap.appiot.ThingTypeToThingTemplateAssociation">
            <End EntitySet="ThingTypes" Role="ThingType"></End>
            <End EntitySet="ThingTemplates" Role="ThingTemplate"></End>
        </AssociationSet>
        <AssociationSet Name="PropertySetToThingTypeAssociation"
Association="com.sap.appiot.PropertySetToThingTypeAssociation">
            <End EntitySet="PropertySets" Role="PropertySet"></End>
            <End EntitySet="ThingTypes" Role="ThingType"></End>
        </AssociationSet>
        <AssociationSet Name="PropertySetToDescriptionAssociation"
Association="com.sap.appiot.PropertySetToDescriptionAssociation">
            <End EntitySet="PropertySets" Role="PropertySet"></End>
            <End EntitySet="Descriptions" Role="Description"></End>
        </AssociationSet>
        <AssociationSet Name="PropertySetTypeToAnnotationAssociation"
Association="com.sap.appiot.PropertySetTypeToAnnotationAssociation">
            <End EntitySet="PropertySetTypes" Role="PropertySetType"></
End>
            <End EntitySet="Annotations" Role="Annotation"></End>
        </AssociationSet>
        <AssociationSet Name="PropertySetTypeToPropertyAssociation"
Association="com.sap.appiot.PropertySetTypeToPropertyAssociation">
            <End EntitySet="PropertySetTypes" Role="PropertySetType"></
End>
            <End EntitySet="Properties" Role="Property"></End>
        </AssociationSet>
        <AssociationSet Name="PropertyTypeToDescriptionAssociation"
Association="com.sap.appiot.PropertyTypeToDescriptionAssociation">

```

```

        <End EntitySet="PropertyTypes" Role="PropertyType"></End>
        <End EntitySet="Descriptions" Role="Description"></End>
    </AssociationSet>
    <AssociationSet Name="PackageToDescriptionAssociation"
Association="com.sap.appiot.PackageToDescriptionAssociation">
        <End EntitySet="Packages" Role="Package"></End>
        <End EntitySet="Descriptions" Role="Description"></End>
    </AssociationSet>
    <AssociationSet Name="ThingTypeToObjectAssociation"
Association="com.sap.appiot.ThingTypeToObjectAssociation">
        <End EntitySet="ThingTypes" Role="ThingType"></End>
        <End EntitySet="ObjectAssociations"
Role="ObjectAssociation"></End>
    </AssociationSet>
    <AssociationSet Name="AnnotationToDescriptionAssociation"
Association="com.sap.appiot.AnnotationToDescriptionAssociation">
        <End EntitySet="Annotations" Role="Annotation"></End>
        <End EntitySet="Descriptions" Role="Description"></End>
    </AssociationSet>
    <AssociationSet Name="PackageToPropertyTypeAssociation"
Association="com.sap.appiot.PackageToPropertyTypeAssociation">
        <End EntitySet="Packages" Role="Package"></End>
        <End EntitySet="PropertyTypes" Role="PropertyType"></End>
    </AssociationSet>
    <AssociationSet Name="TemplatePropertiesToNamedPST"
Association="com.sap.appiot.TemplatePropertiesToNamedPST">
        <End EntitySet="DYN_ENT_TemplateProperties"
Role="DYN_ENT_TemplateProperties"></End>
        <End EntitySet="DYN_ENT_NamedPST" Role="DYN_ENT_NamedPST"></
End>
    </AssociationSet>
    <AssociationSet Name="ThingTypeToDescriptionAssociation"
Association="com.sap.appiot.ThingTypeToDescriptionAssociation">
        <End EntitySet="ThingTypes" Role="ThingType"></End>
        <End EntitySet="Descriptions" Role="Description"></End>
    </AssociationSet>
    <AssociationSet Name="PackageToThingTypeAssociation"
Association="com.sap.appiot.PackageToThingTypeAssociation">
        <End EntitySet="Packages" Role="Package"></End>
        <End EntitySet="ThingTypes" Role="ThingType"></End>
    </AssociationSet>
    <AssociationSet Name="PropertyToAnnotationAssociation"
Association="com.sap.appiot.PropertyToAnnotationAssociation">
        <End EntitySet="Properties" Role="Property"></End>
        <End EntitySet="Annotations" Role="Annotation"></End>
    </AssociationSet>
    <AssociationSet Name="ValueToDescriptionAssociation"
Association="com.sap.appiot.ValueToDescriptionAssociation">
        <End EntitySet="Values" Role="Value"></End>
        <End EntitySet="Descriptions" Role="Description"></End>
    </AssociationSet>
    <AssociationSet Name="PackageToThingTemplateAssociation"
Association="com.sap.appiot.PackageToThingTemplateAssociation">
        <End EntitySet="Packages" Role="Package"></End>
        <End EntitySet="ThingTemplates" Role="ThingTemplate"></End>
    </AssociationSet>
    <AssociationSet Name="ThingTemplateToDescriptionAssociation"
Association="com.sap.appiot.ThingTemplateToDescriptionAssociation">
        <End EntitySet="ThingTemplates" Role="ThingTemplate"></End>
        <End EntitySet="Descriptions" Role="Description"></End>
    </AssociationSet>
    <AssociationSet Name="ThingTemplateToTemplateProperties"
Association="com.sap.appiot.ThingTemplateToTemplateProperties">
        <End EntitySet="ThingTemplates" Role="ThingTemplate"></End>
        <End EntitySet="DYN_ENT_TemplateProperties"
Role="DYN_ENT_TemplateProperties"></End>
    </AssociationSet>
</EntityContainer>

```

```

    </Schema>
  </edmx:DataServices>
</edmx:Edmx>

```

## Related Information

[Thing Configuration \[page 352\]](#)

## 8.2.2 Thing Type

Thing type is used to group the property sets that describes a thing.

For example, you can define a thing type Automobiles. You can create different things, for example, different automobiles identified by their license plates or vehicle identification numbers. You can associate properties to a property set type. You can assign a property set type to a thing type as property sets.

**Resource Path:** `http://<server address>[:<port number>][/path]/ThingConfiguration`

**Example for a base URI in a cloud foundry environment:** `https://config-thing-sap.cfapps.eu10.hana.ondemand.com/ThingConfiguration`

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scope
<i>POST</i>	<a href="#">Create a Thing Type [page 374]</a>	<code>/ThingConfiguration/&lt;version&gt;/Packages('&lt;package name&gt;')/ThingTypes</code>	<code>&lt;thingconf&gt;.c</code>
<i>GET</i>	<a href="#">Read Details of a Thing Type [page 379]</a>	<code>/ThingConfiguration/&lt;version&gt;/ThingTypes('&lt;thing type name&gt;')</code>	<code>&lt;thingconf&gt;.r</code>
<i>GET</i>	<a href="#">Read all Thing Types [page 383]</a>	<code>/ThingConfiguration/&lt;version&gt;/ThingTypes</code>	<code>&lt;thingconf&gt;.r</code>
<i>GET</i>	<a href="#">Read Thing Types of a Package [page 387]</a>	<code>/ThingConfiguration/&lt;version&gt;/Packages('package name')/ThingTypes</code>	<code>&lt;thingconf&gt;.r</code>
<i>PATCH</i>	<a href="#">Update a Thing Type [page 390]</a>	<code>/ThingConfiguration/&lt;version&gt;/ThingTypes('&lt;thing type name&gt;')</code>	<code>&lt;thingconf&gt;.u</code>
<i>DELETE</i>	<a href="#">Delete a Thing Type [page 393]</a>	<code>/ThingConfiguration/v1/ThingTypes('&lt;thing type name&gt;')</code>	<code>&lt;thingconf&gt;.d</code>

### Note

- `<thingconf>` refers to the thing configuration application name.
- `<version>` refers to the version of the method, `/v1` or `/v2`

## Thing Type Object Properties

Property	Type	Mandatory	Maximum Length	Description
Name	String	Yes	81	Unique name of the thing type
Description	String	No	60	Description of the thing type for each language with the corresponding ISO language code
<div> <b>i Note</b> <p>If you include <code>Descriptions</code> along with the <code>CopySource</code> field, the method assigns these descriptions for the newly created thing type. If you do not include <code>Descriptions</code> along with the <code>CopySource</code> field, the method copies descriptions defined in the source thing type to the newly created thing type.</p> </div>				
PropertySets	Array of property set objects	No	-	Array of property sets that belong to the thing type
ThingTypeCategory	String	No	Fixed	Category of thing type  Only the following values are supported: <ul style="list-style-type: none"> <li>• <code>ASSET</code> – Indicates the root thing</li> <li>• <code>COMPONENT</code> – Indicates the sub-items of the <code>ASSET</code></li> </ul>
<div> <b>i Note</b> <p>You can define this only using <code>/v2</code> version of the create method.</p> </div>				

Property	Type	Mandatory	Maximum Length	Description
CopySource	String	No	81	<p>Name of the thing type</p> <p>You can use this field to specify the already existing thing type name, structure of which is copied to the new thing type to be created.</p> <div> <p><b>Note</b></p> <ul style="list-style-type: none"> <li>You can use this field only in the /v1 version of the create method.</li> <li>You cannot specify any value other than the thing type name in this field.</li> <li>You cannot create copy of a thing type that belongs to a package with scope=private.</li> <li>In a request payload to copy an existing thing type, you can only specify the name of the new thing type, CopySource field, and Descriptions field.</li> </ul> </div>

## Property Set Object Properties

Property	Type	Mandatory	Maximum Length	Description
Name	String	Yes	50	Unique identifier of the property (that belongs to the propertySetType) defined in a thing type
PropertySetType	String	Yes	81	Name of the property set type referred
Description	String	No	60	Description of the property set for each language with the corresponding ISO language code

### Note

For reasons of data protection and privacy, it is not allowed to assign a property set to a rule context if the property set type to which the property set belongs has been classified as holding personal data, or sensitive data.

Also, if an initially non-classified property set is assigned to a rule context that is used by rules, any later classification of the property set as personal, or sensitive data leads to the deactivation of all affected rules. For more information on creating rule context, see [Create a Rule Context](#).

## Thing Type Description Properties

Property	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the object description
Description	String	255	Description of the object

## Related Information

[Mapping \[page 761\]](#)

[Thing Template \[page 406\]](#)

[Property Set Type \[page 434\]](#)

[Property Type \[page 476\]](#)

[Extended Thing Type \[page 371\]](#)

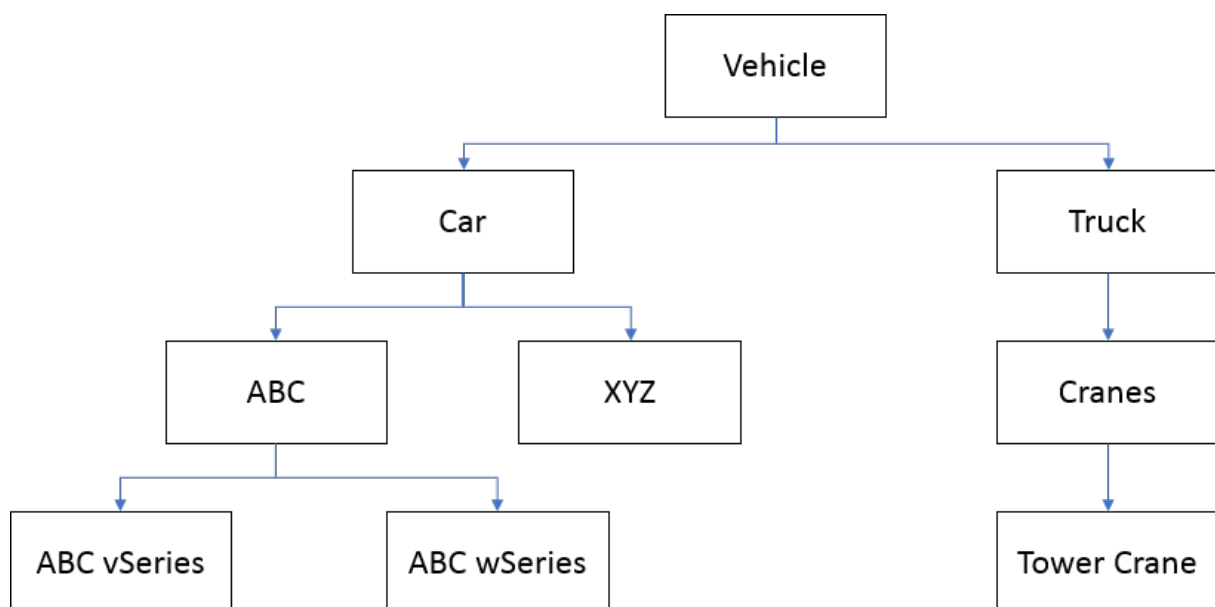
[Thing Type Category \[page 373\]](#)

### 8.2.2.1 Extended Thing Type

In an IoT scenario, multiple thing types may have few of the property sets in common. In such a scenario, you can create a thing type with all the commonly used property sets and then extend it to create child thing types that will inherit all the property sets from the extended thing type. In addition, you can define new property sets for the child thing type. You can extend a thing type within the same package or extend a thing type from a package to which you have access.

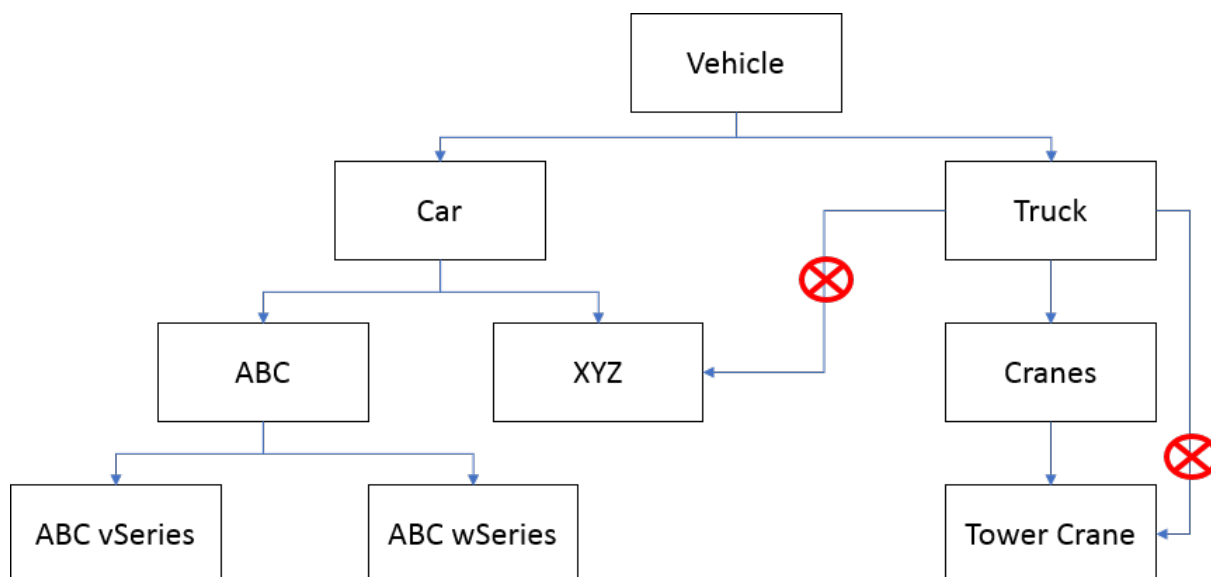
#### Multi-Level Thing Type Extension

Starting with version 1.46, you can extend thing types to multiple levels using OData service. The following diagram illustrates the multi-level extension of thing types:



In the above example, **Vehicle** is the extended thing type. The child thing types **Car** and **Truck** inherits all property sets from the extended thing type.

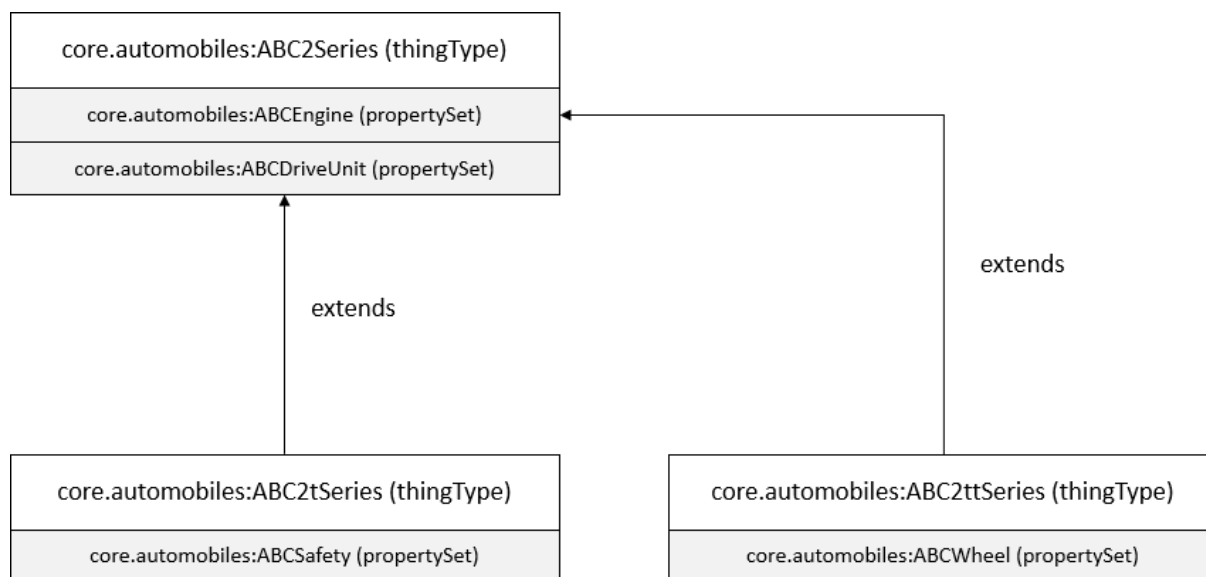
The following diagram illustrates the multiple thing type extension and the cyclic dependency, which is not supported:



In the above example,

- The child thing type **XYZ** cannot inherit the property sets from both the extended thing types **Car** and **Truck**.
- You cannot create a cyclic dependency by extending the thing type **Truck** to create the child thing type **Tower Crane**.

Here is another example of extended thing type and the inheritance of its property sets.



As the example in the diagram shows, there are two thing types with a specialized property set each, dealing with safety and wheel properties, respectively. Both thing types refer to the same base thing type providing



property sets dealing with engine and drive unit properties. As a result of the extension, the two child thing types have the following property sets:

- `core.automobiles:ABC2tSeries`
  - `core.automobiles:Safety`
  - `core.automobiles:Engine`
  - `core.automobiles:DriveUnit`
- `core.automobiles:ABC2ttSeries`
  - `core.automobiles:Wheel`
  - `core.automobiles:Engine`
  - `core.automobiles:DriveUnit`

#### → Recommendation

For performance reasons, we recommend to extend the thing types up to a maximum of five levels.

#### i Note

The property sets across the child thing type and the extended thing type must be unique.

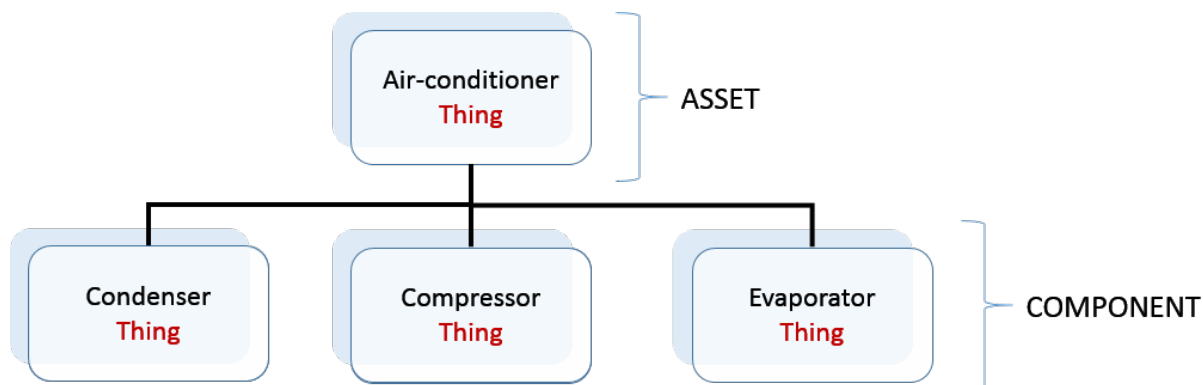
## 8.2.2.2 Thing Type Category

### Thing Type Category

A thing in the Internet of Things (IoT) scenario, may refer to multiple connected things. For example, an air-conditioner is used to monitor and regulate the air temperature, which is achieved by multiple internal parts of the air-conditioner, such as condenser, compressor, evaporator, and so on. The condenser facilitates heat transfer and the compressor pressurizes the refrigerant gas as part of the process to turn it back into a liquid. To depict this example in a thing model scenario, we can categorize the different parts of the air-conditioner in a hierarchical structure in the definition of a thing type. The following are the supported thing type categories:

- **ASSET** – Indicates the root thing (thing of things). In the example of an air-conditioner, the air-conditioner is an asset.
- **COMPONENT** – Indicates the sub-items of the asset. In the example of an air-conditioner, the condenser, compressor, and evaporator are the components.

The following diagram illustrates the thing model with thing type categories:



#### i Note

You can define thing type category for a thing type only using /v2 version of the create method.

### 8.2.2.3 Create a Thing Type

With this method, you create thing types in a specific package.

#### i Note

In the configuration, you can define any number of thing types. However, you can create only one thing type per `POST` request. You can create more number of thing types for a package through multiple `POST` requests.

#### → Recommendation

You can define up to 10 thing types for a package.

#### Naming convention for thing type

- Always prefixed with the package name.
- Can be a combination of alphanumeric characters and underscore.
- Each name must be fully qualified and contains a package name, colon as a separator and the object name. Maximum length of fully qualified name is 81 characters.
- Do not start the object name with a digit or an underscore.
- Length should be at least 3 characters.
- Camel case is generally allowed, however there is a uniqueness check carried out against textual case conflicts:
  - Two names with same characters of different textual case is not allowed. For example: A value "AbcDef" is not accepted when there is another value "abcDEF" that exists.
- Verify that a colon is used only to separate a package name and the object name. The syntax is: `<package>:<name>`. For example: `core.automobiles:FlyWheel`
- Name must be unique within a package.

#### Naming convention for property set

- Only the following characters are supported:
  - Uppercase and lowercase alphabets `a` through `z` and `A` through `Z`
  - Numeric digits `0` through `9`
  - Punctuation marks underscore (`_`), hyphen (`-`), colon (`:`), and full-stop (`.`)
- Maximum length is 50 characters.
- Property set names have to be unique for a given set of thing types.
- Property sets must be unique for a thing type. You can create multiple property sets with unique names for a thing type using the same property set type.

## Request

### URI

Version	URI
v1	/ThingConfiguration/v1/ Packages ('<package name>')/ThingTypes
v2	/ThingConfiguration/v2/ Packages ('<package name>')/ThingTypes

#### i Note

Thing types created using the /v1 version are accessible using the /v2 version, and vice versa. However, the only difference between the /v1 and /v2 versions is that you can define and retrieve thing type category for a thing type only using /v2 version.

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** <thingconf>.c

### Request Headers

Header	Required	Values
Accept-Language	No	Language of the thing type description. The default language is EN.

### Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	string	Unique name of the package

### Request Example

```
/ThingConfiguration/v1/Packages ('core.automobiles')/ThingTypes
```

## Payload

**Format:** *JSON*

#### Example 1

```
/ThingConfiguration/v1/Packages ('core.automobiles')/ThingTypes
```

Creates the thing type `core.automobiles:ABCXSeries` in the package `core.automobiles`.

```
{
  "Name": "core.automobiles:ABCXSeries",
  "Descriptions": [
    {
```

```

        "LanguageCode": "en",
        "Description": "ABCXSeries electric car"
    },
    {
        "LanguageCode": "de",
        "Description": "ABCXSeries Elektroauto"
    }
],
"PropertySets": [
    {
        "Name": "CarEngine",
        "PropertySetType": "core.automobiles:Engine",
        "Descriptions": [
            {
                "LanguageCode": "en",
                "Description": "Car Engine Properties"
            },
            {
                "LanguageCode": "de",
                "Description": "Automotoreigenschaften"
            }
        ]
    },
    {
        "Name": "CarStyle",
        "PropertySetType": "core.automobiles:Style",
        "Descriptions": [
            {
                "LanguageCode": "en",
                "Description": "Car style properties"
            },
            {
                "LanguageCode": "de",
                "Description": "Fahrzeugstileigenschaften"
            }
        ]
    }
]
}

```

## Example 2

/ThingConfiguration/v1/Packages('core.automobiles')/ThingTypes

Creates the thing type `core.automobiles:ABC2Series` which is extended from the thing type `core.automobiles:ABCXSeries` in the package `core.automobiles`.

```

{
    "Name": "core.automobiles:ABC2Series",
    "ExtendedThingType": "core.automobiles:ABCXSeries",
    "Descriptions": [
        {
            "LanguageCode": "en",
            "Description": "ABC2Series electric car"
        },
        {
            "LanguageCode": "de",
            "Description": "ABC2Series Elektroauto"
        }
    ],
    "PropertySets": [
        {
            "Name": "wheel",
            "PropertySetType": "core.automobiles:Wheel",
            "Descriptions": [
                {
                    "LanguageCode": "en",

```

```

        "Description": "Car wheel properties"
      },
      {
        "LanguageCode": "de",
        "Description": "Felgen Eigenschaften"
      }
    ]
  }
]
}

```

### Example 3

/ThingConfiguration/v2/Packages('core.automobiles')/ThingTypes

Creates the thing type `core.automobiles:ABCXSeries` of category [ASSET](#).

```

{
  "Name": "core.automobiles:ABCXSeries",
  "ThingTypeCategory": "ASSET",
  "Descriptions": [
    {
      "LanguageCode": "en",
      "Description": "ABCXSeries electric car"
    },
    {
      "LanguageCode": "de",
      "Description": "ABCXSeries Elektroauto"
    }
  ],
  "PropertySets": [
    {
      "Name": "CarEngine",
      "PropertySetType": "core.automobiles:Engine",
      "Descriptions": [
        {
          "LanguageCode": "en",
          "Description": "Car Engine Properties"
        },
        {
          "LanguageCode": "de",
          "Description": "Automotoreigenschaften"
        }
      ]
    }
  ]
}

```

### Example 4

/ThingConfiguration/v1/Packages('core.automobiles')/ThingTypes

Creates the thing type `core.automobiles:ABCYSeries` copied from the already existing thing type `core.automobiles:ABCXSeries`. The method copies the entire structure of the source thing type to the new thing type. However, you can include the descriptions of the thing type in the `Descriptions` field along with the `CopySource` field. In which case, the method creates new thing type with these descriptions instead of copying it from the source thing type.

```

{
  "Name": "core.automobiles:ABCYSeries",
  "CopySource": "core.automobiles:ABCXSeries"
}

```

## i Note

Ensure that the source thing type belongs to a package with scope=tenant as you cannot create copy of a thing type that belongs to a package with scope=private.

## Response

### Response Status and Error Codes

Code	Description
201	Thing type created successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples.

The following is the response payload after creating a thing type `core.automobiles:ABCXSeries`.

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABCXSeries')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABCXSeries')",
      "type": "com.sap.apiot.ThingType"
    },
    "Name": "core.automobiles:ABCXSeries",
    "PackageName": "core.automobiles",
    "Description": "",
    "ExtendedThingType": null,
    "PropertySets": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABCXSeries')/PropertySets"
      }
    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABCXSeries')/Descriptions"
      }
    },
    "ThingTemplates": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABCXSeries')/ThingTemplates"
      }
    }
  }
}
```

```

    }
  }
}

```

The following is the response payload after creating the thing type `core.automobiles:ABC2Series` which is extended from the thing type `core.automobiles:ABCXSeries`.

```

{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')",
      "type": "com.sap.apptot.ThingType"
    },
    "Name": "core.automobiles:ABC2Series",
    "PackageName": "core.automobiles",
    "Description": "",
    "ExtendedThingType": "core.automobiles:ABCXSeries",
    "PropertySets": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/PropertySets"
      }
    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/Descriptions"
      }
    },
    "ThingTemplates": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/ThingTemplates"
      }
    }
  }
}

```

## 8.2.2.4 Read Details of a Thing Type

With this method, you can retrieve details of a thing type with a specific name.

### Request

**URI:** `/ThingConfiguration/v1/ThingTypes('<Name>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

## Request Headers

Header	Required	Values
Accept-Language	No	Language of the thing type description. The default language is <a href="#">en</a> .

## Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	string	Name of the thing type

## Query String Parameters

Parameter	Required	Data Type	Description
\$expand	No	string	Includes the related resources inline while retrieving the requested resources.  The associations of a thing type that can be retrieved using \$expand are Description, PropertySet, and Properties of PropertySet.

## Request Example

```
/ThingConfiguration/v1/ThingTypes('core.automobiles:ABC2Series')
```

Retrieves details of the thing type `core.automobiles:ABC2Series`

```
/ThingConfiguration/v1/ThingTypes('core.automobiles:ABC2Series')?
$expand=PropertySets/Properties
```

Retrieves the property sets along with the properties for the thing type `core.automobiles:ABC2Series`

## Response

### Response Headers

Header	Description
Etag	Entity tag of the newly created thing type instance. The system generates a new value and replaces the old value every time a thing type is updated.

### Response Status and Error Codes

Code	Description
200	Details of a specific thing type retrieved successfully



## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as depicted in the following examples:

### Example 1

/ThingConfiguration/v1/ThingTypes('core.automobiles:ABC2Series')

Retrieves details of the thing type core.automobiles:ABC2Series.

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')",
      "type": "com.sap.apptot.ThingType"
    },
    "Name": "core.automobiles%3AABC2Series",
    "PackageName": "core.automobiles",
    "Description": "ABC2Series electric car",
    "ExtendedThingType": null,
    "CopySource": null,
    "PropertySets": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/PropertySets"
      }
    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/Descriptions"
      }
    },
    "ThingTemplates": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/ThingTemplates"
      }
    },
    "ObjectAssociations": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/ObjectAssociations"
      }
    }
  }
}
```

### Example 2

/ThingConfiguration/v1/ThingTypes('core.automobiles:ABC2Series')?  
\$expand=Descriptions

Retrieves details of the thing type `core.automobiles:ABC2Series` along with the detailed list of descriptions.

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')",
      "type": "com.sap.apptot.ThingType"
    },
    "Name": "core.automobiles%3AABC2Series",
    "PackageName": "core.automobiles",
    "Description": "",
    "ExtendedThingType": null,
    "CopySource": null,
    "PropertySets": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/PropertySets"
      }
    },
    "Descriptions": {
      "results": [
        {
          "LanguageCode": "en",
          "Description": "ABC2Series electric car"
        },
        {
          "LanguageCode": "de",
          "Description": "ABC2Series Elektroauto"
        }
      ]
    },
    "ThingTemplates": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/ThingTemplates"
      }
    },
    "ObjectAssociations": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/ObjectAssociations"
      }
    }
  }
}
```

### Example 3

```
/ThingConfiguration/v1/ThingTypes('core.automobiles:ABC2Series')?
$expand=PropertySets
```

Retrieves details of the thing type `core.automobiles:ABC2Series` with the detailed list of property sets.

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')",
```

```

        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/
ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')",
        "type": "com.sap.apiot.ThingType"
    },
    "Name": "core.automobiles%3AABC2Series",
    "PackageName": "core.automobiles",
    "Description": "Vehicle thing type",
    "ExtendedThingType": null,
    "CopySource": null,
    "PropertySets": {
        "results": [
            {
                "Name": "engine",
                "ThingType": "core.automobiles%3AABC2Series",
                "Description": "",
                "PropertySetType": "core.automobiles:Engine",
                "DataCategory": "TimeSeriesData",
                "Operation": ""
            },
            {
                "Name": "car",
                "ThingType": "core.automobiles%3AABC2Series",
                "Description": "",
                "PropertySetType": "core.automobiles:Car",
                "DataCategory": "TimeSeriesData",
                "Operation": ""
            }
        ]
    },
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/
Descriptions"
        }
    },
    "ThingTemplates": {
        "__deferred": {
            "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/
ThingTemplates"
        }
    },
    "ObjectAssociations": {
        "__deferred": {
            "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/
ObjectAssociations"
        }
    }
}

```

## 8.2.2.5 Read all Thing Types

With this method, you can retrieve all thing types.

This method retrieves all thing types to which the user has access authorization. It includes thing types from all packages from the same tenant.

In addition, you can retrieve a subset of all thing types, according to the filter criteria provided. For a collective request for a set of thing types, you have these options:

- You can restrict the set of thing types to be retrieved by filter criteria based on the thing type name, description, and extended thing type name.
- You can define that the set of matching thing types shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ThingConfiguration/v1/ThingTypes`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

## Request Headers

Header	Required	Values
<code>Accept-Language</code>	No	Language of the thing type description. The default language is <a href="#">en</a> .

## Query Request Parameters

Parameter	Required	Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set.
<code>\$skip</code>	No	Integer	Number of the first n records to be excluded from the result set.
<code>\$inlinecount</code>	No	String	Total number of thing types available; permissible values are <a href="#">allpages</a> and <a href="#">none</a> . The following results apply if this parameter is used in conjunction with the other query parameters: <ul style="list-style-type: none"><li>• If you use this with the <code>\$top</code> and <code>\$skip</code> parameters, the result contains the total count of thing types available.</li><li>• If you use this with <code>\$filter</code> parameter, the result contains the total count of thing types based on the defined filter condition.</li></ul>

Parameter	Required	Type	Description
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; only valid field is the thing type name.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$filter	No	String	Filter condition to be applied; the only valid fields are thing type name, description, and extended thing type name. The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> .  Multiple filters are supported using the separators – <a href="#">and</a> , <a href="#">or</a> .  You can use the string functions such as <a href="#">substringof</a> , <a href="#">startswith</a> , and <a href="#">endswith</a> to search data using \$filter. Valid fields are thing type name, description, and extended thing type name.
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are thing type name, package name, description, and extended thing type name.

## Request Example

```
/ThingConfiguration/v1/ThingTypes
```

Retrieves all thing types

```
/ThingConfiguration/v1/ThingTypes?$select=Name, PackageName
```

Retrieves all thing types. The result set only contains selected fields - name of the thing type and name of the package.

```
/ThingConfiguration/v1/ThingTypes?$filter=Name eq 'core.automobiles:Vehicle'
```

Retrieves details of a thing type with the specific name.

```
/ThingConfiguration/v1/ThingTypes?$filter=Description eq 'ABC2 Car Series'
```

Retrieves all thing types with a specific description.

## Response

### Response Status and Error Codes

Code	Description
200	Details of all thing types retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

```
{
  "d": {
    "results": [
      {
        "_metadata": {
          "id": "https://thing-configuration-odata-
v010-2003.cfapps.staging.sic.ondemand.com:443/ThingConfiguration/v1/
ThingTypes('core.automobiles.%3ABC2Series')",
          "uri": "https://thing-configuration-odata-
v010-2003.cfapps.staging.sic.ondemand.com:443/ThingConfiguration/v1/
ThingTypes('core.automobiles.%3ABC2Series')",
          "type": "com.sap.apiot.ThingType"
        },
        "Name": "core.automobiles:ABC2Series",
        "PackageName": "core.automobiles",
        "Description": "test thing type",
        "ExtendedThingType": "",
        "PropertySets": {
          "_deferred": {
            "uri": "https://thing-configuration-odata-
v010-2003.cfapps.staging.sic.ondemand.com:443/ThingConfiguration/v1/
ThingTypes('core.automobiles.%3ABC2Series')/PropertySets"
          }
        },
        "Descriptions": {
          "_deferred": {
            "uri": "https://thing-configuration-odata-
v010-2003.cfapps.staging.sic.ondemand.com:443/ThingConfiguration/v1/
ThingTypes('core.automobiles.%3ABC2Series')/Descriptions"
          }
        }
      }
    ],
    "_next": "https://thing-configuration-odata-
v010-2003.cfapps.staging.sic.ondemand.com:443/ThingConfiguration/v1/ThingTypes?
$format=json&$skiptoken=20&$filter=Name eq 'core.automobiles:ABC2Series'"
  }
}
```

## 8.2.2.6 Read Thing Types of a Package

With this method, you can retrieve all thing types or a specific thing type of a package with specific ID.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

### Request

**URI:** `/ThingConfiguration/v1/Packages('<PackageName>')/ThingTypes`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

### Request Headers

Header	Required	Values
Accept-Language	No	Language of the thing type description. The default language is <a href="#">en</a> .

### Request Parameters

Parameter	Required	Data Type	Description
PackageName	Yes	String	Unique name of the package

### Query Request Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$inlinecount	No	String	<p>Total number of thing types available; permissible values are <a href="#">allpages</a> and <a href="#">none</a>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"><li>• If you use this with the <code>\$top</code> and <code>\$skip</code> parameters, the result contains the total count of thing types available.</li><li>• If you use this with <code>\$filter</code> parameter, the result contains the total count of thing types based on the defined filter condition.</li></ul>

Parameter	Required	Data Type	Description
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; valid fields are name.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$filter	No	String	Filter condition to be applied; the only valid fields are thing type name, description, and extended thing type name. The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> .  Multiple filters are supported using the <a href="#">AND</a> separator.  You can use the string functions such as <a href="#">substringof</a> , <a href="#">startswith</a> , and <a href="#">endswith</a> to search data using \$filter. Valid fields are thing type name, description, and extended thing type name.
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are thing type name, package name, description, and extended thing type name.

## Request Example

```
/ThingConfiguration/v1/Packages('core.automobiles')/ThingTypes
```

Retrieves all thing types of the package `core.automobiles`

```
/ThingConfiguration/v1/Packages('core.automobiles')/ThingTypes?$filter=Name eq 'core.automobiles:ABC2Series'
```

Retrieves details of a thing type specified in the filter condition.

```
/ThingConfiguration/v1/Packages('core.automobiles')/ThingTypes?$select=Name, Description
```

Retrieves all thing types of the package. The result set only contains the selected fields, name of the thing type and its description.

```
/ThingConfiguration/v1/Packages('core.automobiles')/ThingTypes?$filter=ExtendedThingType eq 'core.automobiles:ParentTT'
```

Retrieves all thing types that are extended from a specific thing type.



## Response

### Response Status and Error Codes

Code	Description
200	Details of a specific thing type are retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

/ThingConfiguration/v1/Packages('core.automobiles')/ThingTypes

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')",
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')",
        "type": "com.sap.apptot.ThingType"
      },
      "Name": "core.automobiles%3AABC2Series",
      "PackageName": "core.automobiles",
      "Description": "ABC2Series electric car",
      "ExtendedThingType": "",
      "CopySource": null,
      "PropertySets": {
        "__deferred": {
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/PropertySets"
        }
      },
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/Descriptions"
        }
      },
      "ThingTemplates": {
        "__deferred": {
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/ThingTemplates"
        }
      },
      "ObjectAssociations": {
        "__deferred": {
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTypes('core.automobiles%3AABC2Series')/ObjectAssociations"
        }
      }
    ]
  }
}
```

```

    },
    {
      "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTypes('core.automobiles%3AABCXSeries')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTypes('core.automobiles%3AABCXSeries')",
        "type": "com.sap.apptot.ThingType"
      },
      "Name": "core.automobiles%3AABCXSeries",
      "PackageName": "core.automobiles",
      "Description": "ABCXSeries electric car",
      "ExtendedThingType": "",
      "CopySource": null,
      "PropertySets": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTypes('core.automobiles%3AABCXSeries')/PropertySets"
        }
      },
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTypes('core.automobiles%3AABCXSeries')/Descriptions"
        }
      },
      "ThingTemplates": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTypes('core.automobiles%3AABCXSeries')/ThingTemplates"
        }
      },
      "ObjectAssociations": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTypes('core.automobiles%3AABCXSeries')/ObjectAssociations"
        }
      }
    }
  ]
}

```

## 8.2.2.7 Update a Thing Type

With this method, you can to update the specified thing type.

You can update the thing type as follows:

- Add new property sets, edit already existing property sets, or delete the existing property sets
- Add new extended thing type or delete the assigned extended thing type

The **ETag** value is mandatory to update a thing type. You must retrieve the **ETag** value of the thing type using the **GET (/ThingConfiguration/v1/ThingTypes('<thing type name>'))** service. The **ETag** value is assigned to the request header parameter **If-Match** in the **PATCH** request. If you do not include the **ETag** value in the request header or if the **ETag** value is not correct, the server displays an error message.

### Note

You cannot update the thing type name using the `PATCH` request. The thing type name in the request payload is optional. However, if you specify the thing type name in the request payload, it must be the same as the one specified in the request URL.

After updating a thing type, the system generates a new `ETag` value for the thing type and replaces the old value with the new value.

### Note

- You cannot change the name of the thing type using `PATCH` request.
- You can add a new extended thing type, only if the thing type to be updated is not an extension of another thing type. If you are not editing the existing extended thing type, you must still specify the extended thing type details in the request payload. If you do not specify the details, the system deletes the extended thing type. If the thing type to be updated is already an extension of another thing type, you can change the extended thing type. For example, assume that the thing type **ABC** is an extension of the thing type **Car**. When you update the thing type **ABC**, you can remove the extended thing type **Car** and add a new extended thing type **Vehicle**.
- The valid operations are `ADD`, `DELETE`, and `UPDATE`.
  - You cannot delete the property sets if they are inherited from the parent thing type. You can only delete the property set from the parent thing type.
  - You cannot delete the property sets if data exists for the child thing type that has inherited this property set.

## Request

**URI:** `/ThingConfiguration/v1/ThingTypes('<thing type name>')`

**Operation Type:** `CRUD`

**HTTP Method:** `PATCH`

**Permissions:** `<thingconf>.u`

### Request Headers

Header	Required	Values
<code>Accept-Language</code>	No	Language of the thing type description. The default language is <a href="#">en</a> .
<code>If-Match</code>	Yes	Latest <code>ETag</code> value of a specific thing type that is updated using the <code>PATCH</code> request.

### Request Parameters

Parameter	Required	Data Type	Description
<code>thing type name</code>	Yes	String	Name of the thing type

## Request Example

```
/ThingConfiguration/v1/ThingTypes('core.automobiles:ABC2Series')
```

## Response

### Response Headers

Name	Description
Etag	Identifies the updated thing type

### Response Status and Error Codes

Code	Description
204	Thing type updated successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples:

### Example 1

```
/ThingConfiguration/v1/ThingTypes('core.automobiles:ABC2Series')
```

The following request payload is used to add a new property set for the thing type `core.automobiles:ABC2Series`.

```
{
  "Name": "core.automobiles:ABC2Series",
  "PropertySets": [
    {
      "Name": "ABCCarWheel",
      "PropertySetType": "core.automobiles:Wheel",
      "Descriptions": [
        {
          "LanguageCode": "en",
          "Description": "Car wheel properties"
        }
      ],
      "Operation": "ADD"
    }
  ]
}
```

### Example 2

```
/ThingConfiguration/v1/ThingTypes('core.automobiles:ABC2Series')
```

The following request payload is used to change the extended thing type from `core.automobiles:ABCSeries` to `core.automobiles:XYZSeries` and add a new property set for the thing type `core.automobiles:ABC2Series`.

```
{
  "ExtendedThingType": "core.automobiles:XYZSeries",
  "Descriptions": [],
  "PropertySets": [
    {
      "Name": "wheel_fr",
      "PropertySetType": "core.automobiles:Wheel",
      "Descriptions": [
        {
          "LanguageCode": "en",
          "Description": "Wheel Front Right"
        }
      ],
      "Operation": "ADD"
    }
  ]
}
```

## 8.2.2.8 Delete a Thing Type

With this method, you can delete a thing type with a specific name.

The `ETag` value is mandatory to delete a thing type. You must retrieve the `ETag` value of the thing type using the `GET (/ThingConfiguration/v1/ThingTypes('<thing type name>'))` service. The `ETag` value is assigned to the request header parameter `If-Match` in the `PATCH` request. If you do not include the `ETag` value in the request header, the server sends HTTP response code 428. If the `ETag` value is not correct, the server sends HTTP response code 412.

### Note

- You cannot delete a thing type if one of the following reasons is true:
  - Extended to another thing type
  - Things are already created
- If you delete a thing type which has been mapped to one or more sensor types, the system automatically deletes all mappings referring to that thing type.

## Request

**URI:** `/ThingConfiguration/v1/ThingTypes('<thing type name>')`

**Operation Type:** CRUD

**HTTP Method:** `DELETE`

**Permissions:** `<thingconf>.d`

## Request Headers

Header	Required	Values
If-Match	Yes	Latest ETag value of a specific thing type that is updated using the PATCH request.

## Request Parameters

Parameter	Required	Data Type	Description
thing type name	Yes	String	Unique name of the thing type

## Request Example

```
/ThingConfiguration/v1/ThingTypes('core.automobiles:ABC2Series')
```

## Response

### Response Status and Error Codes

Code	Description
204	Thing type deleted successfully

## 8.2.3 Thing Type Association

Association between thing types

### Overview

Association is a way of grouping objects, where every object is related to one or more objects in a unique way. You can create association between thing types, which in turn enables you to create association between thing instances of the associated thing types. However, the scope of the association between thing types depends on the scope of the package to which the thing types belong.

For example, a **Truck** carries multiple **Cars**. The thing type **Truck** has 1:N association with the thing type **Car** as illustrated in the following diagram.



A **Bogie** is part of an **Engine**. The thing type **Bogie** has 1:1 association with the thing type **Engine** as illustrated in the following diagram.



## Types of Association

The following are the predefined association types:

- onetomany
- onetoone
- manytomany
- manytoone

## Types of Relationship

The following are the predefined relationship types:

- **IS A**: Car is a type of vehicle. You can associate the thing type **Car** with the thing type **Vehicle** and assign the **IS A** relationship.
- **HAS A**: Car has wheels. You can associate the thing type **Wheel** with the thing type **Car** and assign the **HAS A** relationship.
- **PART OF**: Bogie is part of the engine. You can associate the thing type **Bogie** with the thing type **Engine** and assign the **PART OF** relationship.

### OData Version: 2.0

**Root URI:** `://<server address>[:<port number>][<path>]/ThingConfiguration`

**Example for a base URI in a cloud foundry environment:** `https://config-thing-sap.cfapps.eu10.hana.ondemand.com/ThingConfiguration`

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scope
POST	<a href="#">Create Thing Type Association [page 397]</a>	<code>/v1/ThingType('&lt;thing type name')/ObjectAssociations</code>	<code>&lt;thingconf&gt;.c</code>
GET	<a href="#">Read Association Details of a Thing Type [page 400]</a>	<code>/v1/ThingType('&lt;thing type name')/ObjectAssociations</code>	<code>&lt;thingconf&gt;.r</code>

HTTP Method	Operation	URI	Scope
DELETE	<a href="#">Delete Thing Type Association [page 405]</a>	/v1/ ObjectAssociations(P rimaryEntity='<thing type name>',AssociatedEn tity='<thing type name>',Relationship ='<relationship name>')	<thingconf>.d

## Thing Type Association Object Properties

Property	Type	Mandatory	Maximum Length	Description
AssociatedEntity	String	Yes	81	Thing type associated with the primary thing type
Description	String	No	60	Description of the thing type association for each language with the corresponding ISO language code
AssociationType	String	Yes	20	Type of association between the primary thing type and the associated thing type
Relationship	String	Yes	50	Type of relationship between the primary thing type and the associated thing type
ExpiryTime	TimeStamp	No	50	Expiry time after which the association between the thing types becomes invalid
CreationTime	TimeStamp	No	50	Time when the association is created, which is the current time

## Thing Type Association Description Properties

Property	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the object description



Property	Data Type	Maximum Length	Description
Description	String	255	Description of the object

### 8.2.3.1 Create Thing Type Association

With this service, you can create association between thing types based on the predefined relationship and association types. The expiry time of the thing type association must be greater than the current time. The association is valid until the defined expiry time, after which, the association is invalid. To extend the validity of the association, you can delete the associated entity and recreate the association with a different expiry time. If you do not specify the expiry time, the association is valid indefinitely. You can only associate thing types that belong to the same package.

#### i Note

The primary thing type and the associated thing type must have unique association type and relationship. For example, the thing type **ACBXSeries** is associated with the primary thing type **Truck** with an expiry date **28-10-2018**. The relationship is **IS A** and the association type is **onetoone**. You cannot create another association between the thing types **Truck** and **ABCXSeries** with relationship **IS A** and association type **manytoone**.

## Request

**URI:** `/ThingConfiguration/v1/ThingTypes('<thing type name>')/ObjectAssociations`

**Operation Type:** CRUD

**HTTP Method:** [POST](#)

**Permissions:** `<thingconf>.c`

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
Name	Yes	String	Unique name of the thing type which represents the primary thing type	Path

## Request Example

`/ThingConfiguration/v1/ThingTypes('core.automobiles:Truck')/ObjectAssociations`

## Payload

**Format:** [JSON](#)

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples:

### Example 1

/ThingConfiguration/v1/ThingTypes('core.automobiles:Truck')/ObjectAssociations

In the below example payload, the thing type `core.automobiles:ABCXSeries` and `core.automobiles:ABC2Series` are associated with the primary thing type `core.automobiles:Truck`. The relationship and association type are `IS A` and `onetoone`, respectively. At the same time, you can create an association between the thing types `core.automobiles:Truck` and `core.automobiles:ABCXSeries` with different relationship and association type as illustrated in Example 2.

```
{
  "Associations": [
    {
      "AssociatedEntity": "core.automobiles:ABCXSeries",
      "Descriptions": [
        {
          "LanguageCode": "en",
          "Description": "Association between Truck and ABCXSeries"
        }
      ],
      "AssociationType": "onetoone",
      "Relationship": "IS A",
      "ExpiryTime": "2018-10-16T08:26:56.229Z"
    }, {
      "AssociatedEntity": "core.automobiles:ABC2Series",
      "Descriptions": [
        {
          "LanguageCode": "en",
          "Description": "Association between Truck and ABC2Series"
        }
      ],
      "AssociationType": "onetoone",
      "Relationship": "IS A",
      "ExpiryTime": "2018-10-16T08:26:56.229Z"
    }
  ]
}
```

### Example 2

/ThingConfiguration/v1/ThingTypes('core.automobiles:Truck')/ObjectAssociations

```
{
  "Associations": [
    {
      "AssociatedEntity": "core.automobiles:ABCXSeries",
      "Descriptions": [
        {
          "LanguageCode": "en",
          "Description": "Association between Truck and ABCXSeries"
        }
      ],
      "AssociationType": "manytoone",
      "Relationship": "HAS A",
      "ExpiryTime": "2018-10-16T08:26:56.229Z"
    }
  ]
}
```

## Response

### Response Status and Error Codes

Code	Description
201	Association created successfully

## Payload

The following is the response payload after creating the thing type association.

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ObjectAssociations(PrimaryEntity='core.automobiles%3ATruck',AssociatedEntity='core.automobiles%3AABCXSeries',Relationship='PART%20OF')",
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ObjectAssociations(PrimaryEntity='core.automobiles%3ATruck',AssociatedEntity='core.automobiles%3AABCXSeries',Relationship='PART%20OF')",
          "type": "com.sap.apptot.ObjectAssociation"
        },
        "AssociationEntityType": "ThingType",
        "PrimaryEntity": "core.automobiles:Truck",
        "PrimaryTenant": "TenantP",
        "AssociatedEntity": "core.automobiles:ABCXSeries",
        "AssociationType": "onetoone",
        "Relationship": "IS A",
        "AssociationExpiryTime": "2018-10-16T08:26:56.229Z",
        "AssociationCreationTime": "2017-12-13T05:17:37.640Z",
        "AssociatedTenant": "TenantP",
        "Description": "association between Truck and ABCXSeries",
        "Descriptions": {
          "__deferred": {
            "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ObjectAssociations(PrimaryEntity='core.automobiles%3ATruck',AssociatedEntity='core.automobiles%3AABCXSeries',Relationship='PART%20OF')/Descriptions"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ObjectAssociations(PrimaryEntity='core.automobiles%3ATruck',AssociatedEntity='core.automobiles%3AABC2Series',Relationship='PART%20OF')",
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ObjectAssociations(PrimaryEntity='core.automobiles%3ATruck',AssociatedEntity='core.automobiles%3AABC2Series',Relationship='PART%20OF')",
          "type": "com.sap.apptot.ObjectAssociation"
        },
        "AssociationEntityType": "ThingType",
        "PrimaryEntity": "core.automobiles:Truck",
        "PrimaryTenant": "TenantP",
        "AssociatedEntity": "core.automobiles:ABC2Series",
        "AssociationType": "onetoone",
        "Relationship": "IS A",
        "AssociationExpiryTime": "2018-10-16T08:26:56.229Z",
        "AssociationCreationTime": "2017-12-13T05:17:37.640Z",
        "AssociatedTenant": "TenantP",
        "Description": "association between Truck and ABC2Series",
        "Descriptions": {
          "__deferred": {
            "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ObjectAssociations(PrimaryEntity='core.automobiles%3ATruck',AssociatedEntity='core.automobiles%3AABC2Series',Relationship='PART%20OF')/Descriptions"
          }
        }
      }
    ]
  }
}
```

```

        "type": "com.sap.apptot.ObjectAssociation"
      },
      "AssociationEntityType": "ThingType",
      "PrimaryEntity": "core.automobiles:Truck",
      "PrimaryTenant": "TenantP",
      "AssociatedEntity": "core.automobiles:ABC2Series",
      "AssociationType": "onetoone",
      "Relationship": "IS A",
      "AssociationExpiryTime": "2018-10-16T08:26:56.229Z",
      "AssociationCreationTime": "2017-12-13T05:17:37.640Z",
      "AssociatedTenant": "TenantP",
      "Description": "association between Truck and ABC2Series",
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ObjectAssociations(PrimaryEntity='core.automobiles
%3ATruck',AssociatedEntity='core.automobiles%3AABC2Series',Relationship='PART
%20OF')/Descriptions"
        }
      }
    }
  ]
}

```

## Related Information

[Thing Type Association \[page 394\]](#)

### 8.2.3.2 Read Association Details of a Thing Type

With this service, you can retrieve all thing types associated with the specified thing type. In addition, you can retrieve the association details of a specific associated thing type.

#### Request

**URI:** /ThingConfiguration/v1/ThingTypes('<thing type name>')/ObjectAssociations

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <thingconf>.r

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
Name	Yes	String	Unique name of the thing type which represents the primary thing type	Path

## Query String Parameters

Parameter	Required	Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$inlinecount	No	String	<p>Total number of thing types available; permissible values are <i>allpages</i> and <i>none</i>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"><li>• If you use this with the \$top and \$skip parameters, the result contains the total count of thing types available.</li><li>• If you use this with \$filter parameter, the result contains the total count of thing types based on the defined filter condition.</li></ul>

Parameter	Required	Type	Description
\$filter	No	String	<p>Filter condition to be applied; the only valid fields are <code>AssociatedEntity</code>, <code>Relationship</code>, and <code>AssociationType</code>. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the separators – <i>and</i>, <i>or</i>.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using \$filter. Valid fields are <code>AssociatedEntity</code>, <code>Relationship</code>, and <code>AssociationType</code>.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are <code>AssociatedEntity</code>, <code>AssociationType</code>, <code>Relationship</code>, <code>AssociationCreationTime</code>, <code>AssociatedTenant</code>, <code>AssociationExpiryTime</code>, <code>Description</code>, and <code>AssociationEntityType</code>.</p>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; valid fields are <code>AssociatedEntity</code>, <code>AssociationType</code>, <code>Relationship</code>, <code>AssociationExpiryTime</code>, and <code>AssociationCreationTime</code>.</p> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>

## Request Example

```
/ThingConfiguration/v1/ThingTypes('core.automobiles:Truck')/ObjectAssociations
```

## Response

### Response Status and Error Codes

Code	Description
200	Association details retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples:

### Example 1

```
/ThingConfiguration/v1/ThingTypes('core.automobiles:Truck')/ObjectAssociations
```

Retrieves all the associated thing types of the primary thing type `core.automobiles:Truck`

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ObjectAssociations(ParentEntity='core.automobiles
%3ATruck',AssociatedEntity='core.automobiles%3AABCXSeries',Relationship='IS
%20A')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ObjectAssociations(ParentEntity='core.automobiles
%3ATruck',AssociatedEntity='core.automobiles%3AABCXSeries',Relationship='IS
%20A')",
        "type": "com.sap.apptot.ObjectAssociation"
      },
      "AssociationEntityType": "ThingType",
      "PrimaryEntity": "core.automobiles:Truck",
      "PrimaryTenant": "TenantP",
      "AssociatedEntity": "core.automobiles:ABCXSeries",
      "AssociationType": "onetoone",
      "Relationship": "IS A",
      "AssociationExpiryTime": "2018-10-16T08:26:56.229Z",
      "AssociationCreationTime": "2017-12-12T10:18:37Z",
      "AssociatedTenant": "TenantP",
      "Description": "association between Truck and ABCXSeries",
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ObjectAssociations(ParentEntity='core.automobiles
%3ATruck',AssociatedEntity='core.automobiles%3AABCXSeries',Relationship='IS
%20A')/Descriptions"
```

```

    }
  },
  {
    "__metadata": {
      "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ObjectAssociations(PrimaryEntity='core.automobiles
%3ATruck',AssociatedEntity='core.automobiles:ABC2Series',Relationship='IS%20A')",
      "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ObjectAssociations(PrimaryEntity='core.automobiles
%3ATruck',AssociatedEntity='core.automobiles:ABC2Series',Relationship='IS%20A')",
      "type": "com.sap.apptot.ObjectAssociation"
    },
    "AssociationEntityType": "ThingType",
    "PrimaryEntity": "core.automobiles%3ATruck",
    "PrimaryTenant": "TenantP",
    "AssociatedEntity": "core.automobiles:ABC2Series",
    "AssociationType": "onetoone",
    "Relationship": "IS A",
    "AssociationExpiryTime": "2018-10-16T08:26:56.229Z",
    "AssociationCreationTime": "2017-12-12T10:18:37Z",
    "AssociatedTenant": "TenantP",
    "Description": "association between Truck and ABC@Series",
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ObjectAssociations(PrimaryEntity='core.automobiles
%3ATruck',AssociatedEntity='core.automobiles%3AABC2Series',Relationship='IS
%20A')/Descriptions"
      }
    }
  }
]
}
}

```

## Example 2

```

/ThingConfiguration/v1/ThingTypes('core.automobiles:Truck')/ObjectAssociations?
$filter=AssociatedEntity eq 'core.automobiles:ABCXSeries' and Relationship eq 'IS
A'

```

Retrieves details of the specified associated thing type

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ObjectAssociations(PrimaryEntity='core.automobiles
%3ATruck',AssociatedEntity='core.automobiles%3AABCXSeries',Relationship='IS
%20A')",
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ObjectAssociations(PrimaryEntity='core.automobiles
%3ATruck',AssociatedEntity='core.automobiles%3AABCXSeries',Relationship='IS
%20A')",
          "type": "com.sap.apptot.ObjectAssociation"
        },
        "AssociationEntityType": "ThingType",
        "PrimaryEntity": "core.automobiles:Truck",
        "PrimaryTenant": "TenantP",

```



```

        "AssociatedEntity": "core.automobiles:ABCXSeries",
        "AssociationType": "onetoone",
        "Relationship": "IS A",
        "AssociationExpiryTime": "2018-10-16T08:26:56.229Z",
        "AssociationCreationTime": "2017-12-12T10:27:30Z",
        "AssociatedTenant": "TenantP",
        "Description": "association between Truck and ABCXSeries",
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ObjectAssociations(PrimaryEntity='core.automobiles
%3ATruck',AssociatedEntity='core.automobiles%3AABCXSeries',Relationship='IS
%20A')/Descriptions"
            }
        }
    }
}
]
}
}

```

## Related Information

[Thing Type Association \[page 394\]](#)

### 8.2.3.3 Delete Thing Type Association

With this service, you can delete the association between the specified thing types.

The ETag value is mandatory to delete association of a thing type. You must retrieve the ETag value of the associated thing type using the GET (/ThingConfiguration/v1/ThingTypes('<thing type name>')/ObjectAssociations?\$filter=AssociatedEntity eq '<thing type name>' and Relationship eq '<relationship>') service. The ETag value is assigned to the request header parameter If-Match in the PUT request. If you do not include the ETag value in the request header or if the ETag value is not correct, the server displays an error message.

You can delete the association between the thing types only if an association is not created between the thing instances belonging to the associated thing types.

## Request

**URI:** /ThingConfiguration/v1/ObjectAssociations(PrimaryEntity='<thing type name>',AssociatedEntity='<thing type name>',Relationship='<relationship name>')

**Operation Type:** CRUD

**HTTP Method:** *DELETE*

**Permissions:** <thingconf>.d

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
Name	Yes	String	Unique name of the primary thing type to which other thing types are associated	Path
Name	Yes	String	Unique name of the thing type which is associated with the primary thing type	Path
Relationship	Yes	String	Unique relationship between the associated thing types	Path

## Request Example

```
/ThingConfiguration/v1/  
ObjectAssociations (PrimaryEntity='core.automobiles:Truck',AssociatedEntity='core.automobiles:ABCXSeries',Relationship='IS A')
```

The thing type `core.automobiles:ABCXSeries` is no more associated with the thing type `core.automobiles:Truck`

.

## Response

### Response Status and Error Codes

Code	Description
204	Associated thing type is no more associated with the primary thing type

## Related Information

[Thing Type Association \[page 394\]](#)

## 8.2.4 Thing Template

### Overview

A thing type is a configuration object and a thing is a data object which is an instance of a thing type. A thing template is a format to maintain property values that remain the same across many thing instances. For example, URL of the thing image, thresholds of thing property values, and model number of a thing remain the same across many things. You can use the thing template to create things and hence use it as a reference to

copy data across things. You can create a thing template based on a thing type. However, the scope of the thing template depends on the scope of the package.

#### OData Version: 2.0

**Root URI:** `http://<server address>[:<port number>][/path]/ThingConfiguration`

**Example for a base URI in a cloud foundry environment:** `https://config-thing-sap.cfapps.eu10.hana.ondemand.com/ThingConfiguration`

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scope
POST	<a href="#">Create a Thing Template [page 409]</a>	<code>/v1/ Packages ('&lt;package name&gt;') / ThingTemplates</code>	<code>&lt;thingconf&gt;.c</code>
GET	<a href="#">Read all Thing Templates for a Package [page 416]</a>	<code>/v1/ Packages ('&lt;package name&gt;') / ThingTemplates</code>	<code>&lt;thingconf&gt;.r</code>
GET	<a href="#">Read all Thing Templates of a Thing Type [page 424]</a>	<code>/v1/ ThingTypes ('&lt;thing type name&gt;') / ThingTemplates</code>	<code>&lt;thingconf&gt;.r</code>
GET	<a href="#">Read Details of a Thing Template [page 427]</a>	<code>/v1/ThingTemplates/ ( '&lt;thing template name&gt;' )</code>	<code>&lt;thingconf&gt;.r</code>
PATCH	<a href="#">Update a Thing Template [page 430]</a>	<code>/v1/ ThingTemplates ('&lt;th ing template name&gt;')</code>	<code>&lt;thingconf&gt;.u</code>
DELETE	<a href="#">Delete a Thing Template [page 432]</a>	<code>/v1/ ThingTemplates ('&lt;th ing template name&gt;')</code>	<code>&lt;thingconf&gt;.d</code>

#### i Note

`<thingconf>` refers to the thing configuration application name.

## Thing Template Object Properties

Property	Type	Mandatory	Maximum Length	Description
<a href="#">Name</a>	String	Yes	81	Unique name of the thing template
<a href="#">Descriptions</a>	String	No	60	Description of the thing template for each language with the corresponding ISO language code
<a href="#">DYN_ENT_TemplateProperties</a>	Array of property set objects	Yes	-	Array of property sets that belong to the thing type
<a href="#">HasFile</a>	Boolean	No	Fixed	Indicates whether to attach a file in the thing template. If the value of this property is true, you can add the content of a file in the <a href="#">fileData</a> property.
<a href="#">FileData</a>		No		<p>Content of the file in encoded format. The content of the file is attached to a thing when you create the thing using the thing template.</p> <p>For example, the image for a thing can be specified in this property in an encoded format. While creating a thing using the thing template, the content (image of the thing) is attached to the thing.</p>

## Related Information

[Thing Type \[page 368\]](#)

## 8.2.4.1 Create a Thing Template

With this method, you can create a thing template for a thing type which belongs to the specified package.

You can create more than one thing template for a thing type and hence, thing template and thing type have a many-to-one relationship. The scope of the thing template is based on the scope of the package to which the thing type belongs. This is because the thing template is created for the thing type.

### i Note

Package name in the request payload is optional. However, if you are specifying the package in the request payload, it must be the same as the one specified in the request URL.

Only the following data categories of property set type are supported in thing templates:

- MasterData
- ReferencePropertyData

### i Note

You can only create a reference property set type for a property set of data category **TimeSeriesData**. However, you cannot directly assign a property set type of data category **ReferencePropertyData** for a thing type. Assume, you create a reference property set type for a property set type of data category **TimeSeriesData** and you assign this property set type for a thing type. When you create a thing template for this thing type, you can assign values for the reference properties. When you create a thing using this thing template with reference property data, the data is copied to the reference properties of the thing.

- Parameters

You can create a thing template only based on the property set data or file data provided in the request payload.

- The `DYN_ENT_TemplateProperties` block in the request payload must contain at least one property set without which the service displays an error message and does not create the template.
- In the `DYN_ENT_TemplateProperties` block of the request payload, if you do not specify the properties for the property set, the service verifies the corresponding property set type. If the properties are defined for the property set type, the service creates a template and assigns null values for the properties.
- In the `DYN_ENT_TemplateProperties` block of the request payload, if the property set specified belongs to a property set type for which you have not defined any properties, the service verifies if the file data is specified in the request payload:
  - If file data is specified, the service creates the thing template.
- If file data is not specified, the service does not create the thing template.

### Naming convention

- Always prefixed with the package name.
- Can be a combination of alphanumeric characters and underscore.
- Each name must be fully qualified and contains a package name, colon as a separator and the object name. Maximum length of fully qualified name is 81 characters.
- Do not start the object name with a digit or an underscore.
- Length should be at least 3 characters.

- Camel case is generally allowed, however there is a uniqueness check carried out against textual case conflicts:
  - Two names with same characters of different textual case is not allowed. For example: A value "AbcDef" is not accepted when there is another value "abcDEF" that exists.
- Verify that a colon is used only to separate a package name and the object name. The syntax is: <package>:<name>. For example: core.automobiles:FlyWheel
- Name must be unique within a package.

## Request

**URI:** /ThingConfiguration/v1/Packages('<package name>')/ThingTemplates

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** <thingconf>.c

### Request Headers

None

### Request Parameters

Parameter	Required	Data Type	Description
package name	Yes	String	Unique identifier of the package

### Request Example

/ThingConfiguration/v1/Packages('core.automobiles:ABC2Series')/ThingTemplates

Request to create thing templates for thing types belonging to the package core.automobiles.

## Payload

**Format:** *JSON*

### Example 1

The following request payload is used to create a thing template core.automobiles:ABC2SeriesTemplate for a thing type core.automobiles:ABC2Series.

```
{
  "Name": "core.automobiles:ABC2SeriesTemplate",
  "ThingType": "core.automobiles:ABC2Series",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Template for ABC2Series electric car"
  }],
  {
    "LanguageCode": "de",
```

```

    "Description": "Vorlage für ABC2Series Elektroauto"
  }],
  "DYN_ENT_TemplateProperties": {
    "car": {
      "Color": "Black"
    }
  }
}

```

## Example 2

The following request payload is used to create a thing template with a file content. For example, you can attach an image for a thing using the file attributes in the request payload to create a thing template.

```

{
  "Name": "core.automobiles:ABC2SeriesTemplate",
  "ThingType": "core.automobiles:ABC2Series",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Template for ABC2Series electric car"
  },
  {
    "LanguageCode": "de",
    "Description": "Vorlage für ABC2Series Elektroauto"
  }],
  "HasFile": true,
  "FileData":
  "iVBORw0KGgoAAAANSUHEUgAAANIAAAACAYAAADigVZlAAAQNO1EQVR4nO2dCXQTxxnH10LT5jVteH1N
+5q+JCKBJITLmHIfKzBHHCCYBAiEw+I2GIMhDQ0kqQolIRc1SV5e
+prmqX3JawQDL64bK8x2Ajb2Bg7NuBjjSXftmRZhyXZ1nZGleL1eGa1kg2iyua9X2TvzvHNN/
Ofb2Z2ZSi04ygZGZm
+EXADZGSCgYAbICMTDATcABmZYCDgBsJIBAMBN0BGJhgIuAEyMsGA1wQdHZ1UV1cX5XK5qM7OzgcMRuNT
rSbTEraq6strhdfzruTk5WpZ8q5c117Jyb6szc3K117RggtFxcWX2dvVB02mtmVOp3NIV2fnQFie2WyB5
QS84TIy/YnXBFB18BMM/pDqat0XzIVM081TSVxyytn6jAuZV4FuzmtzclJz8/
LT8vML0nJzr54HYkpLS88oTkxMMZ48mchlXrxUX1ffCBCUM8xms81Ckgk6pCT6aZvZvCrzYpbu2PfxHAg
8l+obGm0t1vaJQBAPkvI5nM5fWyyWWTU1tFuA+IqOHDvGgehVCK4pA91oGZn
+xlucAC0tthtj4hCT72XOp9S0thi2FBQWPvb13z9RN61QH5s8NYxbMDct7KXyudt7MGeeWLFwrn8iVKz7a
uDZy3Z7dbzz91p43B8ZsjYL1DKmprd3/
ffwpLjWNqbW32xcFuuEyMv2J2M1BJpMpKiExxZKZeamiraltvqvdt8OWL118asq4kNbRzz7NTRo7uuMPo
4Y7Rz/zFbc6411uzHNDuZFDfFe5PICx25/ay2B3bogf/dd9fKCA
+CuytohoskjuylmtLXRwXGuJgy8j0F8QbdrT9bDpzQQ8jSH15+dLt0Vs0Thgzwj7i6Se5kOHDuIljR9mX
RrykjZj/wlVeSONHP8+FhykrJoeOsY8aNoQLAYJa9erShIPvvRsKhQTK/YleX3Pw5K1ErpKt
+iLQjZer6S9IN35VX175r3gw4HU6/Z6ojes/gMKAUQiKBQKiUvvLC1/
MXL18WcKsaZOrJ4WOblY7euUJSOQ7FjZ9Sh2IVC4oLh1hZk6d1LB5/dpt+9R/
hnuq4Xl5VwvT0jLKXS7XOHgaCAm0I2Rk
+gL2oslmewXsiUw5uXlZn8T9LVI5ZWI1jEQTxozkgECgkDrmKqfrFy8ILwJ7om
+3bNoQumTRwtDogE0fTBsf2ggwg+jVBdOCT7eYwGfnti2bQXA6ME2nr9mbnHLOWV/
fEI3WTD00jMzdZjBAKWBwX8ojCgm8vOJoYvLp9qPfHTmy5rXlJ
+BSbtzI5+5EI4ALRCTHHHPaQ8zWqOidO2IooBAKRKRQDwGevJ4w8SQR0e0bmB0QxEKh2IYsdbTW0zmI
xM4/Wi4q9BfQmKcIkCoAEUADgEeI3xOOVEDkicp14e1V2uLwSpTwxNAPwRaGC7OQFqQp9xGDT
+1ksUUubFrMoLFy/VL5g7+4ep48fa
+P0Pz9jnn4H7JCCQBbP79V1rgJDmASE9um7NqvmxMdFbVateiwd7KKswHx
+dwBKwzGq1jgDRrjQ7W5sB6hvsRUHQQCyh8Sg4xwW64/oTpUQ/CIm7xz652yg9flb40R+xIn5i/
LWJKKsk5NOuwqiI7cSQkXooAD6ywE8YneDyLWrDuq/
WR67+BvxcB5dtG9dGHGF7oZsgSuWFz555c0LISKcIvH1AHSDnR0P37h5699pzIW6NrNlptFoIglJ7cOA
gcTf40711nH3g5AguEH3/4YGaZPSj/6Ix/hGmKd/hXQqIanz5q1b8WA5VwOXdlwgoIjAsk2/
Ylv0odUrXj0OT+vgNSCKjgXzZ1eANF3wpI6PRALxcDDT7BlTby+NWPgdqOPBisrKz8E
+zFFXX79Sp9fjhKQiDAqjx6kRHmfCdHDWZek+zCp+gnac6i7XhxOSUkAExiZI7D32y73wtbKfy/
CnPDdEISUkJjsrKiQPhocp86ZPGGeDSzkIWJa1Rq5ccXyDas1X8PBBuG9Cow8UE/
yEaYYPeZybPnFcM1gGRh/+6+KNhNbV1o7Mua29dysrOdblcQ4SvDHmMg5s/I2ZAxNP
+bQz5zaVaAbz0ij7kh6D7NVJnwL1NLJLXn47DCQmXjkXSqAnpFB4/
CO2KkODjEE861B9i7VcKwPlDgaQJQfKi4yFWkNZbPXzZuP4iQRobaLrBiHEpubP0xq2E9989MHnLpg3rX
5hFlz3/1BMcWLaVRm/eeIieNL4KRhi450EjDxQOVaf2T+mrli9bDZaAq3Zu37b3nbf2zvwnwg/d/
DoRENbcYRmhzcn84n5peDkQ0FbNHUmMGjD/LtsGesnC15GEEnYbLH
+c1P9ox6ABiRdKzmDz9ISR0wKgx7WJE7ILtXUUXlQQfGDfTqutC7ch1OUPi8NbPWjZUtBgbIzApFMQhZ
Sccrbrav61zAqWfWR79JbJ8+eG5Q97/HccfB0I/
P4eEJADRigoJP6NBvgzBC715s2coTuwf9+0qi3rKbB3ooCQKCAkCgiJgkKCS7uWfuMbiUkpjzcvCvg9y

```

```

GIkFicwZiGeRMR7oQPB+x8VEy+50cRDiDcoCdBERI/
QsINdmH5pGiPAxUT6cQLxYjky5D7aozdaiQNQ8iLoz
+EhPY1i7FRg7ORKKTUthSDVptTarPZhr737oFHgRj+7lmeVcRsJfrwxdkzc+DSDj50VU6Z0LR5/
drDK5a8HLt4QfhusAfaBUQz8tDHHw/atE5FEhLkods6/ZfHjSDzZWX1JwRCGoxppAbTKG
+gjeadoyZ0Duo43MbU6LmuJpTPCwk3WGFHqTyg9xiJbcIJSS2AtJkWG9R89Imgew8mI9lzmcfQPfeo/
D21iC9wdUZg2oaWoaG7xYvm59vFQ6qHt0EloQycb4WTN25cuttBFBKIRpfAsstkNpVD4Xtye9/802PLFi
/
6J1y6LXpx3mUQleJARHKCaGRbvWLZ01AwQEGUEBIFhOQWDRAS5UVIFOfinrheVHW2MTmFEWgJ1yAVxvFi
KDBlaJA0uJmbrycEcw+3P0PTCDtOeJ1F8uKWCF12fr5EOZzNOL
+g0Qq9Lxz0IQQ7ceUKhSR2jzRxqb2Uj/MP46Ueb2WwyH1hREaPzln+H1FIjY1N+1NSzlirq/
Wfg99/9saunVRszLaHdu3YHg32PueAOP4Klm8lk0JHt4GfZ6yPXE0tf2WxZCHZ7Q7K4XC667I77IuZC5n
ehIRzvBhqJD86s/KgM7CG7p4FUafh8pPsRAeFhu69SfWnjTgBisEi5aKDoQBj17f9FSggWBq/
FPdVSIxIvTh/+Sok3OSI5kf7XbgvR/lyR2REIXV0dIRmX9beys7WljSDzhEeIQFBxFDLX15E7doRMzFs
+ptG+XNmFX726acPHo6Loz45fJhasmihG29CstraqfZ2+wCXyzWCZau
+T0w63d9CQgcy6aACdRxDcJqKkJ9kp9Q9iK9tVGPYqQXgDkbG7wqCX6SgRmyAdmpo7w/
JAYEk1Calj2WgYjOKXL8zsRKFBKNQA4hKp8+c62poaPwjfIOHLOfcX4WAYoqO2jQKLPVSdr+
+azsUkK9Cagdcstnah14rvJ767XdHHSULN64IhISbOdDO9IZYp4gNTIbGd7wCk1ch0jHodf4VJjgKHDig
9nKYNLCDWSQN/3YD6hdWgl38JOLtpA9FTEG4f6JlqwX3pAoJTRMiUgZDKAP1HcyHTrgaYR4xIVFOp/
PJgmUFFfngf52dnU
+Q0nkDLuOsVltb293Cwhib7dTfotlWloaU3slvyANphsUObVDHcISgtlXIwKIzpxSabhlli8zsD
+oJdpGirRS/YIDd4LJeurCTX68WKQsqXA+E9qG
+ho9FSSVIbwnVUGajB1o108xYgKcdLaaouKv6hrNXYOt9ut8P1GAF3hMGWAA83NJVRNpDG4XDcwWg0r
klL27iS0hufgXQDESHhliBCx3oDdUYBIR1LqAOtGxct0DqEHYd7eHg3hMRKbD9D8KvUZ3MqTFuFbVKI
+AidwDh/4soXTj5ouxkabyfJBl
+E5G0f2isfUUjwD5RAzGbzQzWldXOqdbphNbW1VE0NHp1OD6K0TVRI7UCI GusP6Gtq9iWnnOmqu10dhXk
gi3M
+BM5+pNOtELp7pvdWMDRDC4x8B6OzLzrgcLossOPQAcuK2N0XIffXqVI9tqJB5+8Xa7Eu96IuwuP4Suyf0
J85ejhYX0t2MSBTBHh4Vmp4opJYWgxujSZWqr2+ggJAoXY2eAoO/F/CelYYXkVBIMKKB5SJC0sG13rC8/
ALT2fNpzQ6HM9zVW0i4WVXoRP5ZjprufRbB0d0RBfccx0h3v8aCK1voWLTjOE+d/
GsxJELzbAFdPdRMv/KUSwtfX+Es4ulex42kHzGd74Cc8/
ouc8LXen5PV6QD62XEaRXENrrbVI00uIPvMWEhL8F0/37DeSDb4KieRHFpeeKCSdwegGCqmurt4tFn9E
1CMigaWd52/jQX5FulqakprOmMB/LzU3N+OEJNYgKc735agYfbPB16f/pI5jfMgnNVr5UIYPuqxv
+5CXfZ4uAguFgFuKS53hSjQ7UuzrD3x09LYXQ9vN0GQ/k8aOGpe
+T0K6XV1NWaxWKYcNa1sMhgdANHLvgzo7u9zXK1n20PnzaVYQ8ZbB5SFBSPzszkp0vgLjEG
+dyNL4iEBacvBovHQCfIEU42ZWpEP7KiTSS75qifmF/
ss11wc30H3pB1xkEgPJIZKfj5q4yOevkEjix054fgsJfu0BwkcZEQCs3zQ2Ne8pLin5urpad8hkaltQU
LjGbDfimQyLhjg298gDe7tb9Isoabx3wRV0/jXTvgBrfKkE
+aLE8kjzCtcQvD5FB7UCLGyQgh288tTJSEfaVJB68QRQxt/N1GBaRuPmsY/OyP5UYov+DTCvBq65/
JRCGq/Alm3tF
+4xBSzQYncw7VPColhff8ICQqotq7OfRghWKphMZstaxKTUywnTp5qPHP2vOn0mXNcKpNhPpWYxKWmpje
DZd0WtG4vjZORuRcoafEI2QO/
hASXdAajUcozpEGF14uPpgPhWK22xRaLdUbV7eo3b9ws28+yVXsdDvtceHonC0nmPoShey89ien9jkjNL
Qaqrc1MxASw2donpaZn1JeVlyeBfdEv22320/sjMe4DJ8r+GDo7i8K4valKrH8PgsJPkuC
+yL4tgL8JAGPucvKK2MzM7PaWltbl4AyB/wvj10WksZ9CCeCaDSC
+CQkGInq6utF90Q8oIzf5l0tuFheXvkPsi962HN6JwTJ5n6FofEiwn3hsxeShVQF9kVQRPdFSZKwN6Kam
pt3Xiu83mQymcL5a/BrE1BMspBk7kNUd08TveGJoCiShOR+DaiuTvKfFQbpHqmoqMzW6/
WJ8PgbOQ6XkQlKsBd5IUFaDAbJkQhitdpWgKUG226zLYS/y0KS+TGAvdjC3OKmqamFamtroywWq+gpHY/
ZbBnU3GL4FHx
+A8r5BeEhrYxM0BfWA2RkgoGAGyAjEwE3AAZmWAg4AbIyAQDATdARiYYCLgBMjLBQMANKJEJBgJugIxM
MPBfChd6NRZ5pkMAAAAASUVORK5CYII=",
  "DYN_ENT_TemplateProperties": {
    "car": {
      "Color": "Black"
    }
  }
}

```

### Example 3

The following request payload is used to create a thing template `core.automobiles:ABC2SeriesTemplate` for a thing type `core.automobiles:ABC2Series`. Assuming the property set `car` supports multiple languages, you can create data in multiple languages for the property set when you create the thing template. Also, note that only data categories [MasterData](#) and [Parameters](#) support multiple languages for properties of type [String](#).

```
{
```



```

    "Name": "core.automobiles:ABC2SeriesTemplate",
    "ThingType": "core.automobiles:ABC2Series",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Template for ABC2Series electric car"
    },
    {
      "LanguageCode": "de",
      "Description": "Vorlage für ABC2Series Elektroauto"
    }
  ],
  "DYN_ENT_TemplateProperties": {
    "Car": {
      "Color": "{ \"en\": \"Blue\", \"de\": \"Blau\" }"
    }
  }
}

```

#### Example 4

The following request payload is used to create a thing template `core.automobiles:ABC2SeriesTemplate` for a thing type `core.automobiles:ABC2Series`. The request payload includes data for the reference property `pressureUOM`.

```

{
  "Name": "core.automobiles:ABC2SeriesTemplate",
  "ThingType": "core.automobiles:ABC2Series",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Template for ABC2Series electric car"
  },
  {
    "LanguageCode": "de",
    "Description": "Vorlage für ABC2Series Elektroauto"
  }
  ],
  "DYN_ENT_TemplateProperties": {
    "CarEngine": {
      "pressureUOM": "PA"
    }
  }
}

```

#### Note

`carEngine` is the property set that belongs to the property set type of data category **TimeSeriesData**.  
`pressureUOM` is the reference property for the property `pressure`.

## Response

### Response Status and Error Codes

Code	Description
201	Thing template created successfully.

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples:

### Example 1

The following is the response payload after creating the thing template

core.automobiles:ABC2SeriesTemplate:

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3AABC2SeriesTemplate')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3AABC2SeriesTemplate')",
      "type": "com.sap.apptot.ThingTemplate"
    },
    "Name": "core.automobiles%3AABC2SeriesTemplate",
    "ThingType": "core.automobiles:ABC2Series",
    "PackageName": "core.automobiles",
    "HasFile": null,
    "FileData": null,
    "Description": "Template for ABC2Series electric car",
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3AABC2SeriesTemplate')/Descriptions"
      }
    },
    "DYN_ENT_TemplateProperties": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3AABC2SeriesTemplate')/DYN_ENT_TemplateProperties"
      }
    }
  }
}
```

### Example 2

The following is the response payload after creating the thing template

core.automobiles:ABC2SeriesTemplate with file data:

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3AABC2SeriesTemplate')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3AABC2SeriesTemplate')",
      "type": "com.sap.apptot.ThingTemplate"
    },
    "Name": "core.automobiles%3AABC2SeriesTemplate",
    "ThingType": "core.automobiles:ABC2Series",
    "PackageName": "core.automobiles",
    "HasFile": true,
    "FileData":
      "iVBORw0KGgoAAAANSUUEUgAAANIAAAZCAYAAADigVZlAAAQN01EQVR4nO2dCXQTxxnHl0LT5jVteHlN"
  }
}
```

+5q+JCKBJITLmHifKzBHHCCYBAiEw+I2GIMhDQ0kqQolIRc1SV5e  
+prmqX3JawgQDL64bK8x2Ajb2Bg7NuBj jSXftmRZhyXZ1nZGleLleGalkg2iyua9X2TzvzHNN/  
Ofb2Z2ZSi04ygZGZm  
+EXADZGSCgYAbICMTDATcABmZYCDgBs jIBAMBn0BGJhgIuAEyMsGA1wQdHZ1UV1cX5XK5qM7OzgcMRuNT  
rSbTEraq6strhdfzruTk5Wpz8q5c1l17Jyb6szc3K1l17RggtFxcWX2dvVB02mtmVOp3NIV2fnQFie2WYB5  
QS84TIy/YnXBFBi8BMM/pDqat0XzIVM081TSVxyytn6jAuZV4FuzmtzclJz8/  
LT8vML0nJzr54HYkpLS88oTkxMMZ48mchlXrxUXlffBCUM8xms8lCkgk6pCT6aZvZvCrzYpbu2PfxHAg  
8l+obGm0t1vaJQBAPkvI5nM5fWyyWWTU1tfuA+IqOHDvGgehVCK4pA9lOGZn  
+xluCac0thtj4hCT72XOp9S0thi2FBQWPvb13z9RN6lQH5s8NYxbMDct7KXyudt7MGeeWLFrwn8iVKz7a  
uDZy3Z7dbzz91p43B8ZsjYLLDKmprd3/  
ffwpLjWNqbw32xcFuuEyMv2J2M1BJpMpKiExxZKZeamiraltvqvqdt8OWL1l8asq4kNbRzz7NTRo7uuMPo  
4Y7Rz/zFBc641luzHNDuZFDfFe5PICx25/ay2B3bogf/dd9fKCA  
+CuytohOSkjuYLmtLXRWxGuJGy8j0F8QbdrT9bDpzQQ8jSH15+dLt0VsOThgzjw7i6Se5kOHDuIljR9mX  
RrykjZj/wlVeSONHP8+FhykrJoeOsY8aNoQLAYJa9erShIPvvrSxKhQTK/YleX3Pw5KlErpKt  
+iLQjZeR6S9IN35VXl75r3gw4HU6/Z6ojes/gMKAUQiKBQKiUvvLC1/  
MXLl8WcKsaZOrJ4Wobly7euUJsOQ7FjZ9Sh2IVC4oLhinhZk6dlB5/dpt+9R/  
hnuq4Xl5VwvT0jLKXS7XOHgaCAm0I2Rk  
+gL2oslmewXsiUw5uXlZn8T9LVI5ZWI1jEQTxozkgECgkDrmKqfrFy8ILwJ7om  
+3bnQumTrwtDoqE0fTBsf2ggwg+jVBdOCT7eYwGfnti2bQXA6ME2nr9mbnHLOWV/  
fEI3WTD00jMzdZjBAKWbW8ojCqm8v0JoYvLp9qPfHTmy5rXlJ  
+BSbtzI5+5EI4ALRCTHHHpaQ8zWqOidO2IooBAKRKRDDQDwGevJ4w8SQRU0e0bmB0QxEKh2IYsdbTW0zmI  
xM4/Wi4q9BfQMkCikCoAEUADgEeI3xOOVedkicp14e1V2uLwSpTwXNAPwRaGC7OQFqQp9xGDT  
+lksUUubFrMoLFy/VL5g7+4ep48fa  
+P0Pz9jnn4H7JCCQBbP79V1rgJDMASE9um7NqvmxMdfbVateiwd7KKswHx  
+dwBKwzGq1jgDRrjQ7W5sB6hvsRUhQQCyh8Sg4xwW64/oTpUQ/CIm7xz652yg9flb40R+xIn5i/  
LWJKKSK5Nouuqi7cSQkXooAD6ywE8YneDyLwRduq/  
WR67+BvxcB5dtG9dGHGf7oZsgSuWFz555c0LISKcwIvH1AHSdnR0P37h5699pzIW6NrNlptFoIglJ7cOA  
gcTf40711nH3g5AguEH3/4YGaZPSj/6Ix/hGmKd/hXQqIanz5qlb8WA5VwOXDLwgoIjAsk2/  
Ylv0odUrXj00t+vgNSCkjgXzZleANF3wpI6PRALxcDDT7BlTby+NWPgdqQPbisrKz8E  
+zFFXX79Sp9fjHkQIDAqjx6kRHmfCdHDWZek+zCp+gnac6i7XhxOSUkAExiZl7D32y73wtbKfy/  
CnPdDeISUkJjsrKiqPhocp86ZPGGeDSzkIWJa1Rq5ccXyDas1X8PBBuG9Cow8UE/  
yEaYYPeZybPnFcMlGRh/6+KNhNbVl07Mua29dysrOdblcQ4SvDHmMg5s/I2ZAXNP  
+bQz5zaVaABz0ij7kh6D7NVJnwL1NLJLXn47DCQmXjkXsQAnpFB4/  
CO2KKODjEE861B9i7VcKwPldgaJQqfKi4yFWkNZbPXzZuP4iQRobaLrBIhEpubP0xq2E9989MHnLpg3rX  
5hFlz3/1BMcWLaVRm/eeIieNL4KRhi450EjDxQOvAf2T+mrli9bDZaAq3Zu37b3nbf2zvnwg/d/  
DoRENbcYRmhzc84n5peDkQ0FbNHUMMGjD/LtsGesnC5GEEEnYbLH  
+clP9ox6ABiRdKzmDz9ISR0wKgx7WJE7ILtxUuxlQQfGDftQutC7ch1OUPi8NbPWjZUtBgbIzApFMQhZ  
Sccrbrav61zAqWfWR79JbJ8+eG5Q97/HccfBOI/  
P4eEJADRigoJP6NBvgzBC715s2coTuwf9+0qi3rKbB3ooCQKCAkCgiJgkKCS7uWfuMbiUkjpjzcvCvg9y  
GIkFicwZiGERMR7oQPB+x8VEy+5OcrDiDcoCdBERI/  
QsINdmH5pGiPaxUT6cQLxYjky5D7aozdaiQNQ8iLoz  
+EhPYli7FRg7ORKKTUtHSDvptTarPZhr737oFhgrj+7lmeVcRs jfrwxdkzc+DSDj50VU6Z0LR5/  
drDK5a8HLt4QfhUsAfaBUQz8tDHHw/ate5FEHLkods6/ZfHjsdzZWXlJwRCGoxppAbTKG  
+gjeadoyZ0Duo43MbU6LmuJpTPCwk3WGFHqTyg9xiJbcIJSS2AtJkWG9R89Imgew8mi9lzmcfQPfeo/  
D2liC9wdUZg2oaWoaG7xYvm59vFQ6qHt0El0Qyqcb4WTN25cuttBFBKIRpfAsstkNpvd4Xtye9/802PLFi  
/  
6Jly6LXpx3mUQleJARHKCaGrbvWLZO1AwQEGUEBIFhOQWDRAS5UVIFOfinrheVHw2MTmFEWgJlyAVxvFi  
KDBlaJA0uJmbrycEcw+3P0PTCDtOeJlF8uKWCF12fr5EOZzNOL  
+g0Qq9Lxz0IQQ7ceUkhSR2jzRxqb2Uj/MP46Ueb2WwyH1hREaPzln+HlFIjY1N+1NSzliRq/  
Wfg99/9saunVRszLaHdu3YHg32PueAOP4Klm8l0Jht4GfZ6yPXE0tf2WxZCHZ7Q7K4XC667I77IuZC5n  
ehIRzvBhqJD86s/KgM7CG7p4FUafh8pPsRAeFhu69SfWnjTgBisEi5aKDoQBj17f9FSggWBq/  
FPdVSIxIvTh/+Sok3OSI5kf7XbgvR/lyR2REIXV0dIRmX9beys7WljdszhEeIQFBxFDLX15E7doRMzFs  
+pTG+XNmFX726acPHo6Loz45fJhasmihG29CstraqfZ2+wCXyzWCZau  
+T0w63d9CQgcy6aACdRxDCJqKk79kp9Q9iK9tVGPpyQXgDkbg7wqCX6SgRmyAdmpo7w/  
JAYEk1Calj2WgYjOKXL8zsrKFBKNQA4hKp8+c62poaPwjfi0HLOfcX4WAYoq02jQKLPVSdr+  
+azsUkK9CagdCstnah14rvJ767XdhHSUln64IhISbOdDO9IZYp4gNTIbGd7wCk1ch0jHodf4VJjGkHDig  
9nKYNLCDWSQN/3YD6hdWgl38JOLtpA9FTEg4f6Jlqwx3pAoJTRMiUgZDKAP1HcyHTrgaYR4xIVFOp/  
PJgmuFFfngf52dnU  
+Q0nkDLuOsVitlb293Cwhib7dTfotlWloaU3s1vyANpHsUObVDHcISGtlXlIwKIzpxSabhlli8zsD  
+oJdpGirRS/YIDd4LJeurCTX68WKQsqXA+E9qG  
+ho9FSSVibwnVUgajB1ol08xEGKCDLaaouKv6hrNXYOt9ut8PlGAF3hMGWAA83NjVRNpDG4XDcwWg0r  
klLZ7is0hufgXQDESHhliBCx3oDdUYBIRlLqAoTGxct0DqEHYd7eHg3hMRKbD9D8KvUz3MqTFuFbVKI  
+AIdwDh/4soXTj5ouxkabyfJB1  
+E5G0f2isfUUjwD5RAzGbzQzWlDXOqdbphNbW1VE0NHp1OD6KOTVRI7UCIgusP6Gtq9iWnnOmqu10dhXk  
gi3M  
+BM5+pNOtELp7pvDWMRDcC4x8B6OzLzrgcLossOPQAcuK2N0XIffXqVI9tqJB5+8Xa7Eu96IuwuP4Suyf0  
J85ejhYX0t2MSBTBHh4Vmp4opJYWgxujSZWqr2+ggJAoXY2eAo0/F/CeLYYXkVBIMKKB5SjC0sG13rC8/  
ALT2fnPzQ6HM9zVW0i4WVXoRP5ZjprufRbB0d0RBfccx0h3v8aCK1voWLTjOE+d/

```
GsxJEeLzbAFdPdRMv/KUSwtfX+Es4ulex42kHzGd74Cc8/
ouc8LXen5PV6QD62XEaRXENrrbVI00uIPvMWExHl8F0/37DeSdb4KieRHFpeeKCSDwegGCqmurt4tFn9E
1CMigaWd52/jQX5FulqakprOmMB/LzU3N+OEJNYgKc735agYfbPB16f/pI5jfMgnNVr5UiYPuqxV
+5CXFz4uAguFgFuKS53hSQj7UuzrD3x09LYXQ9vN0GQ/k8aOGpe
+T0K6XV1NWaxWKYcNa1sMhgdANHLvgzo7u9zXK1n20PnzaVYQ8ZbB5SFBSPPszskp0vgLjEG
+dyNL4iEBacvBovHqcFIeU42ZWpEP7KiTSS75qifmF/
sS1lw30H3pB1xkEgpJIZKfj5q4yOevkEjix054fgsJfu0BwkcZEgCs3zQ2Ne8pLin5urpad8hkaltQUn
LjGbDfimQyLhjg298gDe7tb9Isoabx3wRV0/jXTvgBrfKkE
+aLE8kjzCtcQvD5FB7UCLgyQgh288tTJSEfaVJB68QRQxt/N1GBaRuPmsY/OyP5UYov+DTCvBq65/
JRCGq/AlM3tF
+4xBSzQYncw7VPCOlhff8ICQqotq7OfRghWKphMZstaxKTUywnTp5qPHP2vOn0mXNcKpNhPpWYxKWmpje
DZd0WtG4vjZORuRcoafEI2QO/
hASXdAajUcozpEGF14uPpgPhWK22xRaLdUbV7eo3b9ws28+yVXsdDvtceHonC0nmPoShey89ien9jkjNL
Qaqrc1MxASw2donpaZn1JeVlyeBfdEv22320/sjMe4DJ8r8+GDo7i8K4valKrH8PgsJPkuC
+yL4tgL8JAGPucvKK2MzM7PaWltbl4AyB/wvj10WksZ9CCeCaDSC
+CQkGInq6utF90Q8oIzf5l0tuFheXvkPsI962HN6JwtJ5n6FofEiwn3hsxeShVQF9kVQRPDfSZKwN6Kam
pt3Xiu83mQymcL5a/BrE1BMspBk7kNUdO8TveGJoCiShOR+DaiuTvKfFQbpHqmoqMzW6/
WJ8PgbOQ6XkQlKsBd5IUfAdAbJkQhitdpWgKUg226zLYS/y0KS+TGAvdj30KmqamFamtroywWq+gpHY/
ZbBnU3GL4FHx
+A8r5BeEhrYxM0BFwA2RkgoGAGyAjEwE3AAZmWAg4AbIyAQDATdARiYYCLgBMjLBQMANKJEJBgJugIxM
MPBfChd6NRZ5pkMAAAAASUVORK5CYII=",
  "Description": "Template for ABC2Series Car",
  "Descriptions": {
    "__deferred": {
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/ThingTemplates('core.automobiles
%3AABC2SeriesTemplate')/Descriptions"
    }
  },
  "DYN_ENT_TemplateProperties": {
    "__deferred": {
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/ThingTemplates('core.automobiles
%3AABC2SeriesTemplate')/DYN_ENT_TemplateProperties"
    }
  }
}
}
```

## Related Information

[Thing Template \[page 406\]](#)

### 8.2.4.2 Read all Thing Templates for a Package

With this method, you can retrieve all thing templates created in the specified package.

You can retrieve a subset of all thing templates, according to the filter criteria provided. For a collective request for a set of thing templates, you have these options:

- You can restrict the set of thing templates to be retrieved by filter criteria based on the thing template name and description.
- You can define that the set of matching thing templates shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records

retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ThingConfiguration/v1/Packages('<package name>')/ThingTemplates`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

## Request Headers

None

## Request Parameters

Parameter	Required	Data Type	Description
<code>package name</code>	Yes	String	Unique identifier of the package

## Query Request Parameters

Parameter	Required	Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set.
<code>\$skip</code>	No	Integer	Number of the first n records to be excluded from the result set.
<code>\$orderby</code>	No	String	Field name to be used for sorting the result set in ascending order; valid fields are thing template name, thing type name, package name, and description.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.

Parameter	Required	Type	Description
\$filter	No	String	<p>Filter condition to be applied; valid fields are thing template name, thing type name, package name, and description. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the separators – <i>and</i>, <i>or</i>.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using <i>\$filter</i>. Valid fields are thing template name and description.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are thing template name, thing type name, package name, and description.</p>

## Request Example

```
/ThingConfiguration/v1/Packages('core.automobiles')/ThingTemplates
```

Retrieves all thing templates created in the package `core.automobiles`.

### Note

If file details are added in the thing template, file attributes *hasFile* and *fileData* are included in the response payload.

## Response

### Response Status and Error Codes

Code	Description
200	Details of all thing templates retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* format for this method has the following structure:

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate')",
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate')",
          "type": "com.sap.apptot.ThingTemplate"
        },
        "Name": "core.automobiles%3AABC2SeriesTemplate",
        "ThingType": "core.automobiles%3AABC2Series",
        "PackageName": "core.automobiles",
        "HasFile": null,
        "FileData": null,
        "Description": "Template for ABC2Series electric car",
        "Descriptions": {
          "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate')/Descriptions"
          }
        },
        "DYN_ENT_TemplateProperties": {
          "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate')/
DYN_ENT_TemplateProperties"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABCXSeriesTemplate')",
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABCXSeriesTemplate')",
          "type": "com.sap.apptot.ThingTemplate"
        },
        "Name": "core.automobiles%3AABCXSeriesTemplate",
        "ThingType": "core.automobiles%3AABCXSeries",
        "PackageName": "core.automobiles",
        "HasFile": null,
        "FileData": null,
        "Description": "Template for ABCXSeries electric ar",
        "Descriptions": {
          "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABCXSeriesTemplate')/Descriptions"
          }
        },
        "DYN_ENT_TemplateProperties": {
          "__deferred": {
```

```

        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABCXSeriesTemplate')/
DYN_ENT_TemplateProperties"
    }
}
]
}
}

```

## Related Information

[Thing Template \[page 406\]](#)

### 8.2.4.3 Read all Thing Templates

With this method, you can retrieve all thing templates.

You can retrieve a subset of all thing templates, according to the filter criteria provided. For a collective request for a set of thing templates, you have these options:

- You can restrict the set of thing templates to be retrieved by filter criteria based on the thing template name, thing type name, package name, and description.
- You can define that the set of matching thing templates shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ThingConfiguration/v1/ThingTemplates`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

## Request Headers

None

## Request Parameters

None



## Query Request Parameters

Parameter	Required	Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; valid fields are thing template name, thing type name, package name, and description.</p> <p>To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; valid fields are thing template name, thing type name, package name, and description. The filter operators supported are <a href="#">eq</a>, <a href="#">lt</a>, <a href="#">gt</a>, <a href="#">le</a>, <a href="#">ge</a>, and <a href="#">ne</a>.</p> <p>Multiple filters are supported using the separators – <a href="#">and</a>, <a href="#">or</a>.</p> <p>You can use the string functions such as <a href="#">substringof</a>, <a href="#">startswith</a>, and <a href="#">endswith</a> to search data using <code>\$filter</code>. Valid fields are thing template name and description.</p>
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are thing template name, thing type name, package name, and description.

## Request Example

/ThingConfiguration/v1/ThingTemplates

Retrieves all thing templates.

## i Note

If file details are added in the thing template, file attributes *hasFile* and *fileData* are included in the response payload.

## Response

### Response Status and Error Codes

Code	Description
200	Details of all thing templates retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* format for this method has the following structure:

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3Atemplatel')",
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3Atemplatel')",
          "type": "com.sap.apptot.ThingTemplate"
        },
        "Name": "core.automobiles:templatel",
        "ThingType": "core.automobiles:ABCXSeries",
        "PackageName": "core.automobiles",
        "HasFile": null,
        "FileData": null,
        "Description": "Description in English",
        "Descriptions": {
          "__deferred": {
            "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3Atemplatel')/Descriptions"
          }
        },
        "DYN_ENT_TemplateProperties": {
          "__deferred": {
            "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3Atemplatel')/DYN_ENT_TemplateProperties"
          }
        }
      },
      {
        "__metadata": {
```

```

        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.computers%3Atemplate2')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.computers%3Atemplate2')",
        "type": "com.sap.apptot.ThingTemplate"
    },
    "Name": "core.computers:template2",
    "ThingType": "core.computers:CompXYZ",
    "PackageName": "core.computers",
    "HasFile": null,
    "FileData": null,
    "Description": "Description in English",
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.computers%3Atemplate2')/Descriptions"
        }
    },
    "DYN_ENT_TemplateProperties": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.computers%3Atemplate2')/DYN_ENT_TemplateProperties"
        }
    }
},
{
    "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.fourwheelers%3Atemplate3')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.fourwheelers%3Atemplate3')",
        "type": "com.sap.apptot.ThingTemplate"
    },
    "Name": "core.fourwheelers:template3",
    "ThingType": "core.fourwheelers:carTT",
    "PackageName": "core.fourwheelers",
    "HasFile": null,
    "FileData": null,
    "Description": "Description in English",
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.fourwheelers%3Atemplate3')/Descriptions"
        }
    },
    "DYN_ENT_TemplateProperties": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.fourwheelers%3Atemplate3')/DYN_ENT_TemplateProperties"
        }
    }
}
]
}
}

```

## Related Information

[Thing Template \[page 406\]](#)

### 8.2.4.4 Read all Thing Templates of a Thing Type

With this method, you can retrieve all thing templates created for a thing type.

You can retrieve a subset of all thing templates, according to the filter criteria provided. For a collective request for a set of thing templates, you have these options:

- You can restrict the set of thing templates to be retrieved by filter criteria based on the thing template name and description.
- You can define that the set of matching thing templates shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ThingConfiguration/v1/ThingTypes('<thing type name>')/ThingTemplates`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

## Request Headers

None

## Request Parameters

Parameter	Required	Data Type	Description
<code>thing type name</code>	Yes	String	Name of the thing type

## Query Request Parameters

Parameter	Required	Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set.

Parameter	Required	Type	Description
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; valid fields are thing template name, thing type name, package name, and description.</p> <p>To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; valid fields are thing template name, thing type name, package name, and description. The filter operators supported are <a href="#">eq</a>, <a href="#">lt</a>, <a href="#">gt</a>, <a href="#">le</a>, <a href="#">ge</a>, and <a href="#">ne</a>.</p> <p>Multiple filters are supported using the separators – <a href="#">and</a>, <a href="#">or</a>.</p> <p>You can use the string functions such as <a href="#">substringof</a>, <a href="#">startswith</a>, and <a href="#">endswith</a> to search data using <code>\$filter</code>. Valid fields are thing template name and description.</p>
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are thing template name, thing type name, package name, and description.

## Request Example

```
/ThingConfiguration/v1/ThingTypes('core.automobiles:ABC2Series')/ThingTemplates
```

Retrieves all thing templates created for the thing type `core.automobiles:ABC2Series`.

### Note

If file details are added in the thing template, file attributes [hasFile](#) and [fileData](#) are included in the response payload.

## Response

### Response Status and Error Codes

Code	Description
200	Details of all thing templates retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* format for this method has the following structure:

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate')",
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate')",
          "type": "com.sap.apptot.ThingTemplate"
        },
        "Name": "core.automobiles%3AABC2SeriesTemplate",
        "ThingType": "core.automobiles%3AABC2Series",
        "PackageName": "core.automobiles",
        "HasFile": null,
        "FileData": null,
        "Description": "Template for ABC2Series electric car",
        "Descriptions": {
          "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate')/Descriptions"
          }
        },
        "DYN_ENT_TemplateProperties": {
          "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate')/
DYN_ENT_TemplateProperties"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate2')",
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate2')",
          "type": "com.sap.apptot.ThingTemplate"
        },
        "Name": "core.automobiles%3AABC2SeriesTemplate2",
        "ThingType": "core.automobiles%3AABC2Series",

```

```

        "PackageName": "core.automobiles",
        "HasFile": null,
        "FileData": null,
        "Description": "Template 2 for ABC2Series electric car",
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate2')/Descriptions"
            }
        },
        "DYN_ENT_TemplateProperties": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
ThingTemplates('core.automobiles%3AABC2SeriesTemplate2')/
DYN_ENT_TemplateProperties"
            }
        }
    }
}
]
}
}

```

## Related Information

[Thing Template \[page 406\]](#)

### 8.2.4.5 Read Details of a Thing Template

With this method, you can retrieve details of the specified thing template.

#### Request

**URI:** /ThingConfiguration/v1/ThingTemplates('<thing template name>')

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <thingconf>.r

#### Request Headers

None

## Request Parameters

Parameter	Required	Data Type	Description
thing template name	Yes	String	Name of the thing template

## Request Example

/ThingConfiguration/v1/ThingTemplates('core.automobiles:ABC2SeriesTemplate')

Retrieves the details of the thing template `core.automobiles:ABC2SeriesTemplate`.

### Note

If file details are added in the thing template, file attributes *hasFile* and *fileData* are included in the response payload.

## Response

### Response Headers

Header	Description
Etag	Entity tag of the newly created thing template instance. The system generates a new value and replaces the old value every time a thing template is updated.

### Response Status and Error Codes

Code	Description
200	Details of thing template retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* format for this method has the following structure:

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3AABC2SeriesTemplate')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/ThingTemplates('core.automobiles%3AABC2SeriesTemplate')",
      "type": "com.sap.apptot.ThingTemplate"
    },
    "Name": "core.automobiles%3AABC2SeriesTemplate",
    "ThingType": "core.automobiles:ABC2Series",
    "PackageName": "core.automobiles",
    "HasFile": true,
    "FileData":
      "iVBORw0KGgoAAAANSUUEUgAAANIAAAZCAYAAADigVZlAAQ01EQVR4nO2dCXQTxxnH10LT5jVteH1N
```



+5q+JCKBJITLmHIFkzBHHCCYBAiEw+I2GIMhDQ0kqQolIRc1SV5e  
+prmqX3JawgQDL64bK8x2Ajb2Bg7NuBjJxSftmRZhyXZ1nZGleLleGalKg2iyua9X2TzvzHNN/  
Ofb2Z2ZSi04ygZGZm  
+EXADZGSCgYAbICMTDATcABmZYCDgBsJIBAMBN0BGJhgIuAEyMsGA1wQdHZ1UV1cX5XK5qM7OzgcMRuNT  
rSbTEraq6strhdfzruTk5Wpz8q5c1l17Jyb6szc3K1l17RggtFxcWX2dvVB02mtmVOp3NIV2fnQFie2WYB5  
QS84TIy/YnXBFBi8BMM/pDqat0XzIVM081TSVxyytn6jAuZV4FuzmtzclJz8/  
LT8vML0nJzr54HYkpLS88oTkxMMZ48mchlXrxUXlffBCUM8xms8lCkgk6pCT6aZvZvCrzYpbu2PfxHAg  
8l+obGm0t1vaJQBAPkvI5nM5fWyyWWTU1tFuA+IqOHDvGgehVCK4pA9lOGzn  
+xluCac0thtj4hCT72XOp9S0thi2FBQWPvb13z9RN6lQH5s8NYxbMDct7KXyudt7MGeeWLFrwn8iVKz7a  
uDZy3Z7dbzz91p43B8ZsjYLLDKmprd3/  
ffwpLjWNqbw32xcFuuEyMv2J2M1BJpMpKiExxZKZeamiralrtvvqdt8OWL1l8asq4kNbRzz7NTRo7uuMPo  
4Y7Rz/zFBc641luzHNDuZFDfFe5PICx25/ay2B3bogf/dd9fKCA  
+CuytohOSkjuYlmtLXRwXGuJGy8j0F8QbdrT9bDpzQQ8jSH15+dLt0VsOThgzjw7i6Se5kOHDuIljR9mX  
RrykjZj/wlVeSONHP8+FhykrJoeOsY8aNoQLAYJa9erShIPvvrSxKhQTK/YleX3Pw5KlErpKt  
+iLQjZeR6S9IN35VXl75r3gw4HU6/Z6ojes/gMKAUQiKBQKiUvvLC1/  
MXLl8WcKsaZOrJ4Wobly7euUJsOQ7FjZ9Sh2IVC4oLhnhZk6d1LB5/dpt+9R/  
hnuq4Xl5VwvT0jLKXS7XOHgaCAm0I2Rk  
+gL2oslmewXsiUw5uXlZn8T9LVI5ZWI1jEQTxozkgECgkDrmKqfrFy8ILwJ7om  
+3bnQumTrwtDoqE0fTBsf2ggwg+jVBdOCT7eYwGfnti2bQXA6ME2nr9mbnHLOWV/  
fEI3WTD00jMzdZjBAKWbW8ojCqm8v0JoYvLp9qPfHTmy5rXlJ  
+BSbtzI5+5EI4ALRCTHHHPaQ8zWqOidO2IooBAKRKRDDwGevJ4w8SQR0e0bmB0QxEKh2IYsdbTW0zmI  
xM4/Wi4q9BfQMkCikCoAEUADgEeI3xOOVedkicp14e1V2uLwSpTwxNAPwRaGC7OQFqQp9xGDT  
+1ksUUubFrMoLFy/VL5g7+4ep48fa  
+P0Pz9jnn4H7JCCQBbP79V1rgJDMASE9um7NqvmxMdfbVateiwd7KKswHx  
+dwBKwzGq1jgDRrjQ7W5sB6hvsRUhQQCyh8Sg4xwW64/oTpUQ/CIm7xz652yg9flb40R+xIn5i/  
LWJKKSK5Nouuqi7cSQkXooAD6ywE8YndyLwRduq/  
WR67+BvxcB5dtG9dGHGf7oZsgSuWFz555c0LISKcwIvH1AHSdnR0P37h5699pzIW6NrNlptFoIglJ7COA  
gcTf40711nH3g5AguEH3/4YGaZPSj/6Ix/hGmKd/hXQqIanz5q1b8WA5VwOXDLwgoIjAsk2/  
Ylv0odUrXj00t+vgNSCkjgXzZleANF3wpI6PRALxcDDT7BlTby+NWPGdqQPbisrKz8E  
+zFFXX79Sp9fjHkQIDAqjx6kRHmfCdHDWZek+zCp+gnac6i7XhxOSUkAExiZl7D32y73wtbKfy/  
CnPDDeISUkJjsrKiqPhocp86ZPGGeDSzkIWJa1Rq5ccXyDas1X8PBBuG9Cow8UE/  
yEaYYPeZybPnFcMlGRh/6+KNhNbVl07Mua29dysrOdblcQ4SvDHmMg5s/I2ZAXNP  
+bQz5zaVaABz0ij7kh6D7NVJnwL1NLJLXn47DCQmXjkXsQAnpFB4/  
CO2KKODjEE861B9i7VcKwPldgaJQqfKi4yFWkNZbPXzZuP4iQRobaLrBIhEpubP0xq2E9989MHnLpg3rX  
5hFlz3/1BMcWLaVRm/eeIieNL4KRhi450EjDxQOvAf2T+mrli9bDZaAq3zu37b3nbf2zvnwg/d/  
DoRENbcYRmhzc84n5peDkQ0FbNHUMMGjD/LtsGesnC5GEEEnYbLH  
+clP9ox6ABiRdKzmDz9ISR0wKgx7WJE7ILTxUuxlQQfGDftQutC7ch1OUPi8NbPWjZUtBgbIzApFMQhZ  
Scctrav61zAqWfWR79JbJ8+eG5Q97/HccfBOI/  
P4eEJADRigoJP6NBvgzBC715s2coTuwf9+0qi3rKbB3ooCQKCAkCgiJgkKCS7uWfuMbiUkjpzpcvcvCvg9y  
GIkfiCwZiGERMR7oQPB+x8VEy+5OcrDiDcoCdBERI/  
QsINdmH5pGiPaxUT6cQLxYjky5D7aozdaiQNQ8iLoz  
+EhPY1i7FRg7ORKKTUtHSDvptTarPZhr737oFhgrj+7lmeVcRsjfrwxdkzc+DSDj50VU6Z0LR5/  
drDK5a8HLt4QfhUsAfaBUQz8tDHHw/ate5FEHLkods6/ZfHjsdzZWXlJwRCGoxppAbTKG  
+gjeadoyZ0Duo43MbU6LmuJpTPCwk3WGFHqTyg9xiJbcIJSS2AtJkWG9R89Imgew8mi9lzmcfQPfeo/  
D21iC9wdUZg2oaWoaG7xYvm59vFQ6qHt0El0Qyqcb4WTN25cuttBFBKIRpfAsstkNpvd4Xtye9/802PLFi  
/  
6Jly6LXpx3mUQleJARHKCaGrbvWLZO1AwQEGUEBIFhOQWDRAS5UVIFOfinrheVHw2MTmFEWgJlyAVxvFi  
KDBlaJA0uJmbrycEcw+3P0PTCDtOeJlF8uKWCF12fr5EOZzNOL  
+g0Qq9Lxz0IQQ7ceUkhSR2jzRxqb2Uj/MP46Ueb2WwyH1hREaPzln+HlFIjY1N+1NSzliRq/  
Wfg99/9saunVRszLaHdu3YHg32PueAOP4Klm8l0Jht4GfZ6yPXE0tf2WxZCHZ7Q7K4XC667I77IuZC5n  
ehIRzvBhqJD86s/KgM7CG7p4FUafh8pPsRAeFhu69SfWnjTgBisEi5aKDoQBj17f9FSggWBq/  
FPdVSIxIvTh/+Sok3OSI5kf7XbgvR/lyR2REIXV0dIRmX9beys7WljdszhEeIQFBxFDLX15E7doRMzFs  
+pTG+XNmFX726acPHo6Loz45fJhasmihG29CstraqfZ2+wCXyzWCZau  
+T0w63d9CQgcy6aACdRxDCJqKk79kp9Q9iK9tVGPpyQXgDkbg7wqCX6SgRmyAdmpo7w/  
JAYEk1Calj2WgYjOKXL8zsrKFBKNQA4hKp8+c62pqaPwjfi0HLOfcX4WAYoq02jQKLPVSdr+  
+azsUkK9CagdCstnah14rvJ767XdhHSU1N64IhISbOdDO9IZYp4gNTIbGd7wCk1ch0jHodf4VJjGkHDig  
9nKYNLCDWSQN/3YD6hdWgl38JOLtpA9FTEg4f6JlqwX3pAoJTRMiUgZDKAP1HcyHTrgaYR4xIVFOp/  
PJgmuFFfngf52dnU  
+Q0nkDLuOsVitlb293Cwhib7dTfotlWloaU3s1vyANpHsUObVDHcISgtlXIwKIzpxSabhlli8zsD  
+oJdpGirRS/YIDd4LJeurCTX68WKQsqXA+E9qG  
+ho9FSSVibwnVUGajB1o108xEGKCDLaaouKv6hrNXYot9ut8PlGAF3hMGWAA83NjVRNpDG4XDcwWg0r  
klLZ7is0hufgXQDESHhliBCx3oDdUYBIRlLqAoTGxct0DQEHYd7eHg3hMRKbD9D8KvUz3MqTFuFbVKI  
+AIdwDh/4soXTj5ouxkabyfJB1  
+E5G0f2isfUUjwD5RAzGbzQzW1dXOqdbphNbW1VE0NHp1OD6KOTVRI7UCIgusP6Gtq9iWnnOmqu10dhXk  
gi3M  
+BM5+pNOtELp7pvDWMRDcC4x8B6OzLzrgcLossOPQAcuK2N0XIffXqVI9tqJB5+8Xa7Eu96IuwuP4Suyf0  
J85ejhYX0t2MSBTBHh4Vmp4opJYWgxujSZWqr2+ggJAoXY2eAo0/F/CeLYYXkVBIMKKB5SJC0sG13rC8/  
ALT2fnPzQ6HM9zVW0i4WVXoRP5ZjprufRbB0dORBfccx0h3v8aCK1voWLTjOE+d/

```
GsxJEeLzbAFdPdRMv/KUSwtfX+Es4ulex42kHzGd74Cc8/
ouc8LXen5PV6QD62XEaRXENrrbVI00uIPvMWExHl8F0/37DeSDb4KieRHFpeeKCSDwegGCqmurt4tFn9E
1CMigaWd52/jQX5FulqakprOmMB/LzU3N+OEJNYgKc735agYfbPB16f/pI5jfMgnNVr5UiYPuqxV
+5CXFz4uAguFgFuKS53hSQj7UuzrD3x09LYXQ9vN0GQ/k8aOGpe
+T0K6XV1NWaxWKYcNA1sMhgdaNHLvgzo7u9zXK1n20PnzaVYQ8ZbB5SFBSPPzszkp0vgLjEG
+dyNL4iEBacvBovHqcFIeU42ZWpEP7KiTSS75qifmF/
ss1lw30H3pB1xkEgpJIZKfj5q4yOevkEjix054fsgJfu0BwkcZEgCs3zQ2Ne8pLin5urpad8hkaltQUn
LjGbDfimQyLhjg298gDe7tb9Isoabx3wRV0/jXTvgBrfKkE
+aLE8kjzCtcQvD5FB7UCLgyQgh288tTJSEfaVJB68QRQXt/N1GBaRuPmsY/OyP5UYov+DTCvBq65/
JRCGq/AlM3tF
+4xBSzQYncw7VPCOlhff8ICQqotq7OfRghWKphMZstaxKTUywnTp5qPHP2vOn0mXNcKpNhPpWYxKWmpje
DZd0WtG4vjZORuRcoafEI2QO/
hASXdAajUcozpEGF14uPpgPhWK22xRaLdUbV7eo3b9ws28+yVXsdDvtceHonC0nmPoShey89ien9jkjNL
Qaqrc1MxASw2donpaZn1JeVlyeBfdEv22320/sjMe4DJ8r8+GDo7i8K4valKrH8PgsJPkuC
+yL4tgL8JAGPucvKK2MzM7PaWltbl4AyB/wvj10WksZ9CCeCaDSC
+CQkGInq6utF90Q8oIzf5l0tuFheXvkPsi962HN6JwTJ5n6FofEiwn3hsxeShVQF9kVQRPDfSZKwN6Kam
pt3Xiu83mQymcL5a/BrE1BMspBk7kNUdO8TveGJoCiShOR+DaiuTvKfFQbpHqmoqMzW6/
WJ8PgbOQ6XkQlKsBd5IUFaDAbJkQhitdpWgKUg226zLYS/y0KS+TGAvdjC3OKmqamFamtroywWq+gpHY/
ZbBnU3GL4FHx
+A8r5BeEhrYxM0BFwA2RkgoGAGyAjEwE3AAZmWAg4AbIyAQDATdARiYYCLgBMjLBQMANKJEJBgJugIxM
MPBfChd6NRZ5pkMAAAAASUVORK5CYII=",
  "Description": "Template for ABC2Series electric car",
  "Descriptions": {
    "__deferred": {
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/ThingTemplates('core.automobiles
%3AABC2SeriesTemplate')/Descriptions"
    }
  },
  "DYN_ENT_TemplateProperties": {
    "__deferred": {
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/ThingTemplates('core.automobiles
%3AABC2SeriesTemplate')/DYN_ENT_TemplateProperties"
    }
  }
}
}
```

## Related Information

[Thing Template \[page 406\]](#)

### 8.2.4.6 Update a Thing Template

With this method, you can update a thing template.

The `ETag` value is mandatory to update a thing template. You must retrieve the `ETag` value of the thing template using the `GET (/ThingConfiguration/v1/ThingTemplates('<thing template name>'))` service. The `ETag` value is assigned to the request header parameter `If-Match` in the `PUT` request. If you do not include the `ETag` value in the request header or if the `ETag` value is not correct, the server displays an error message.

### Note

- You cannot modify a thing template if you have already created things using the template.
- It is mandatory that you must add the `DYN_ENT_TemplateProperties` block when you update the template even if the properties within the block are not updated.

Only the following functions are supported using this service:

- You can change the description of the template.
- You can add new properties in the template.
- You can change the values of the properties in the template.
- You can add, delete, or change the file data for a template.

### Note

You cannot update the thing template name using the `PATCH` request. The thing template name in the request payload is optional. However, if you specify the thing template name in the request payload, it must be the same as the one specified in the request URL.

After updating a thing type, the system generates a new `ETag` value for the thing type and replaces the old value with the new value.

## Request

**URI:** `/ThingConfiguration/v1/ThingTemplates('<thing template name>')`

**Operation Type:** CRUD

**HTTP Method:** `PUT`

**Permissions:** `<thingconf>.u`

### Request Headers

Header	Required	Values
<code>If-Match</code>	Yes	Latest ETag value of a specific thing type that is updated using the <code>PATCH</code> request.

### Request Parameters

Parameter	Required	Data Type	Description
<code>thing template name</code>	Yes	String	Name of the thing template

### Request Example

`/ThingConfiguration/v1/ThingTemplates('core.automobiles:ABC2SeriesTemplate')`

Request to update the details of the thing template `core.automobiles:ABC2SeriesTemplate`. In this example, the description is updated.

## Payload

Format: *JSON*

```
{
  "Name": "core.automobiles:ABC2SeriesTemplate",
  "ThingType": "core.automobiles:ABC2Series",
  "PackageName": "core.automobiles",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Template for ABC Series Car"
  }],
  "DYN_ENT_TemplateProperties": {
    "Car": {
      "RotationSpeed": "10"
    }
  }
}
```

## Response

### Response Headers

Name	Description
Etag	Identifies the updated thing template

### Response Status and Error Codes

Code	Description
200	Thing template updated successfully

## Related Information

[Thing Template \[page 406\]](#)

## 8.2.4.7 Delete a Thing Template

With this method, you can delete a thing template.

The ETag value is mandatory to delete a thing template. You must retrieve the ETag value of the thing template using the GET (/ThingConfiguration/v1/ThingTemplates('<thing template name>')) service. The ETag value is assigned to the request header parameter If-Match in the DELETE request. If you do not include the ETag value in the request header or if the ETag value is not correct, the server displays an error message.

## i Note

You cannot delete a thing template if one of the following is true:

- You have already created things using the thing template.
- File information is included in the thing template and you have created a thing using the template. You must delete the file and the thing before deleting the template.

## Request

**URI:** `/ThingConfiguration/v1/ThingTemplates('<thing template name>')`

**Operation Type:** CRUD

**HTTP Method:** *DELETE*

**Permissions:** `<thingconf>.d`

### Request Headers

Header	Required	Values
If-Match	Yes	Latest ETag value of a specific thing type that is updated using the PUT request.

### Request Parameters

Parameter	Required	Data Type	Description
thing template name	Yes	String	Name of the thing template

### Request Example

`/ThingConfiguration/v1/ThingTemplates('core.automobiles:ABC2SeriesTemplate')`

## Response

### Response Status and Error Codes

Code	Description
200	Thing template deleted successfully

## Related Information

[Thing Template \[page 406\]](#)

## 8.2.5 Property Set Type

Property set type is a group of attributes that are used to describe things.

### Overview

Property set types enable groups of properties to be assigned to a thing. Property set types are assigned to thing types to describe a thing and they are referred as property sets. Property set type can be assigned to one of the available data categories such as [MasterData](#), [TimeSeriesData](#), [DerivedData](#), and [Parameters](#). For example, Car, Wheel, and Engine are valid property set types. The colour and model of the car, pressure and rotation speed of the wheel, and capacity and transmission of an engine are some of the properties that you can define for the corresponding property set type.

The following are the supported predefined annotations for a property set type:

- `com.sap.apptot.security:pii`
  - You can assign this annotation to a property set type that contains data to identify a person or subject. Data such as nick name, birthdate, and email address are few examples that can be categorized as data that identifies a person.
- `com.sap.apptot.security:spi`
  - You can assign this annotation to a property set type that contains critical data of a person. Data such as identification number, ethnic origin, and biometric data are few examples that can be categorized as critical data of a person.

#### Note

For reasons of data protection and privacy, it is not allowed to assign a property set to a rule context if the property set type to which the property set belongs has been classified as holding personal data, or sensitive data.

Also, if an initially non-classified property set is assigned to a rule context that is used by rules, any later classification of the property set as personal, or sensitive data leads to the deactivation of all affected rules. For more information on creating rule context, see [Create a Rule Context](#).

**Resource Path:** `http://<server address>[:<port number>][[/path]]/ThingConfiguration`

**Example for a base URI in a cloud foundry environment:** `https://config-thing-sap.cfapps.eu10.hana.ondemand.com/ThingConfiguration`

### Operations

#### CRUD Operations

HTTP Method	Operation	URI	Scope
POST	<a href="#">Create a Property Set Type [page 442]</a>	<code>/v1/Packages('&lt;package name&gt;')/PropertySetTypes</code>	<code>&lt;thingconf&gt;.c</code>
GET	<a href="#">Read Details of a Property Set Type [page 450]</a>	<code>/v1/PropertySetTypes('&lt;property set type name&gt;')</code>	<code>&lt;thingconf&gt;.r</code>

HTTP Method	Operation	URI	Scope
GET	<a href="#">Read all Property Set Types [page 456]</a>	/v1/PropertySetTypes	<thingconf>.r
GET	<a href="#">Read all Property Set Types of a Package [page 460]</a>	/v1/Packages ('<package name>') /PropertySetTypes	<thingconf>.r
PATCH	<a href="#">Update a Property Set Type [page 464]</a>	/v1/PropertySetTypes ('<property set type name>')	<thingconf>.u
DELETE	<a href="#">Delete a Property Set Type [page 468]</a>	/v1/PropertySetTypes ('<property set type name>')	<thingconf>.d

### Note

<thingconf> refers to the thing configuration application name.

## Property Set Type Object Properties

Property	Type	Mandatory	Maximum Length	Description
Name	string	Yes	81	Name of the property set type
Description	string	No	60	Description of the property set type for each language with the corresponding ISO language code
<div> <div>Note</div> <p>If you include <code>Descriptions</code> along with <code>CopySource</code>, the method assigns these descriptions for the newly created property set type. If you do not include <code>Descriptions</code> along with <code>CopySource</code>, the method copies descriptions defined in the source property set type to the newly created property set type.</p> </div>				
DataCategory	enum	Yes	-	Data category of the property set type

Property	Type	Mandatory	Maximum Length	Description
Properties	Array of properties	No	-	<p>Array of properties of data category <i>MasterData</i> or <i>TimeSeriesData</i></p> <div> <p><b>i Note</b></p> <ul style="list-style-type: none"> <li>For property set type of data categories <i>TimeSeriesData</i> and <i>DerivedData</i>, you can define property types of data type <i>GeoJSON</i>. However, you can create time series data or derived data for a property of type <i>GeoJSON</i> only via data ingestion from SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (Internet of Things Service).</li> </ul> </div> <div> <p><b>→ Recommendation</b></p> <p>For property set type of data category <i>TimeSeriesData</i> or <i>DerivedData</i>, you can define up to 500 properties. However, for better performance, we highly recommend you to define not more than 200 properties per property set type.</p> </div>
ReferredProperty-SetType	string	No	81	<p>Name of the property set type for which you are defining the reference property data.</p> <div> <p><b>i Note</b></p> <p>You can only define a referred property set type for a property set type of data category <i>ReferencePropertyData</i>.</p> </div>



Property	Type	Mandatory	Maximum Length	Description
CopySource	string	No	81	<p>Name of the property set type</p> <p>You can use this property to specify the already existing property set type name, structure of which is copied to the new property set type to be created.</p> <div> <p><b>i Note</b></p> <ul style="list-style-type: none"> <li>You can use this field only in the /v1 version of the create method.</li> <li>You cannot specify any value other than the property set type name in this field.</li> <li>You cannot copy the property set type of data category <i>ReferencePropertyData</i>.</li> <li>You cannot create copy of a property set type that belongs to a package with scope=private.</li> <li>In a request payload to copy an existing property set type, you can only specify the name of the new property set type, CopySource, and Descriptions. All other fields described in this table are not supported.</li> </ul> </div>

Property	Type	Mandatory	Maximum Length	Description
IsMultiLingual	Boolean	No	Fixed	<p>Indicates whether multiple languages are supported for all string type properties of the property set type.</p> <ul style="list-style-type: none"> <li>• If the value of this field is true, you can create data in multiple languages for the string type properties.</li> <li>• If the value of this field is false, you cannot create data in multiple languages for the string type properties.</li> </ul> <p>Properties of data type other than <b>String</b> does not support data in multiple languages.</p> <p>This field is supported only for the data categories <a href="#">MasterData</a> and <a href="#">Parameters</a>.</p>

### Data Category Enumeration Values

The following table describes the list of supported data categories you can use to define the property set type while configuring the package. The different data categories are used to store different types of data.

Value	Description
MasterData	Indicates the property set type for storing master data
DerivedData	Indicates the property set type for storing derived data
TimeSeriesData	Indicates the property set type for storing time series data
Parameters	Indicates the property set type for storing parameter values
ReferencePropertyData	Indicates the property set type for storing reference property data

If the value of the property type changes very frequently, for example, if the data changes for every second or for every hour, it is recommended to use data categories [TimeSeriesData](#) and [DerivedData](#) for the property set type. For example, you could model the speed of a car as [TimeSeriesData](#) and velocity or acceleration as [DerivedData](#). The data categories [Parameters](#) and [MasterData](#) are technically the same. You can model the parameters of a thing as data category [Parameters](#). These parameters do not change often over time. If you have data of category [TimeSeriesData](#), you can derive data of category [DerivedData](#) using data of category [Parameters](#). For example, speed of the car is available as [TimeSeriesData](#) for a specific time interval. You can derive acceleration for a specific time interval as [DerivedData](#) by applying certain [Parameters](#) data on [TimeSeriesData](#) speed of the car.

### Properties Object Properties

Property	Type	Mandatory	Maximum Length	Description
Name	string	Yes	50	Unique name of the property

Property	Type	Mandatory	Maximum Length	Description
Type	enum	Yes	30	<p>Type used by the property</p> <p>Data type <i>LargeString</i> is applicable only for data categories <i>TimeSeriesData</i> and <i>DerivedData</i>.</p>
PropertyLength	string	Yes (For more information on supported data types, refer to <a href="#">Value Data Types for Property Types and Properties [page 1229]</a> )	30	Storage length for the property specified as an integer value
QualityCode	integer	No	1	<p>Indicates if the property has a quality code attached. A value 0 implies there is no quality code attached; value 1 implies that the property has a quality code available.</p> <p>Quality code is supported only for property set type of data category <i>TimeSeriesData</i> and <i>Parameters</i>. You can only attach a quality code for a property of data type <i>Numeric</i>, <i>NumericFlexible</i>, and <i>String</i>.</p> <div> <p><b>i Note</b></p> <p>Only if the property has a quality code attached, an additional property <code>&lt;property_name&gt;_QC</code> is supported for storing the quality code value.</p> </div>

Property	Type	Mandatory	Maximum Length	Description
Description	string	No	60	<p>Description of the property for each language with the corresponding ISO language code</p> <div> <p><b>i Note</b></p> <p>If you do not define a description here and reuse a property type to define the property, the system uses the description defined in the property type.</p> <p>If you define a description here and reuse a property type to define the property, the system uses the description defined here and not the description defined in the property type.</p> </div>
UnitOfMeasure	string	No	5	Indicates the unit of measure in which the property is measured
PropertyType	string	No		You can reuse the property type object to define the properties of the property set type.

Property	Type	Mandatory	Maximum Length	Description
IsMultiLingual	Boolean	No	Fixed	<p>Indicates whether multiple languages are supported for string type properties.</p> <ul style="list-style-type: none"> <li>If the value of this field is true, you can create data in multiple languages for string type properties.</li> <li>If the value of this field is false, you cannot create data in multiple languages for string type properties.</li> </ul>
				<p><b>i Note</b></p> <p>If the property set type to which the string type property belongs does not support multiple languages, then you cannot define the field <b>IsMultiLingual</b> for the property.</p> <p>If you do not want to maintain data in multiple languages for the string type property though the property set type to which the string type property belongs, you must define <b>IsMultiLingual=false</b> for the property.</p>
Annotations	Array of annotations	No		Array of attributes used to qualify properties of a property set type. .

## Properties Annotation Object Properties

Property	Type	Mandatory	Description
Name	String	Yes	Name of the annotation

## Reference Property Type Object Properties

Property	Type	Mandatory	Maximum Length	Description
Name	string	Yes	50	Reference property type identifier defined in a property set type of data category <a href="#">ReferencePropertyData</a>
ReferenceProperty	string	Yes	30	Name of the property type defined in the referred property set type

Property	Type	Mandatory	Maximum Length	Description
AttributeType	string	Yes	81	<p>Attribute type of the reference property type. The following values are supported:</p> <ul style="list-style-type: none"> <li>com.sap.iot.core.LowLowerThreshold</li> <li>com.sap.iot.core.LowerThreshold</li> <li>com.sap.iot.core.UpperThreshold</li> <li>com.sap.iot.core.UpperUpperThreshold</li> <li>com.sap.iot.core.UoM</li> <li>com.sap.iot.core.TargetValue</li> </ul>

#### Property Set Type and Properties Description

Property	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the object description
Description	String	255	Description of the object

## Related Information

[Data Protection and Privacy \[page 322\]](#)

### 8.2.5.1 Create a Property Set Type

With this method, you can create property set types in the specified package.

## Overview

When you create a property set type, you can assign predefined annotations or user-defined annotations for the properties of the property set type. It is optional to provide the ADD operation in the request payload.

However, if you provide the ADD operation for one of the annotations, then you must provide the ADD operation for all the annotations of that property. DELETE operation is not supported for the annotation of a property when you create the property set type.

Based on the scope of the package, the property set type can be reused as follows:

- Only in the thing types within a package, if the scope of the package is private
- Only in the thing types within a tenant, if the scope of the package is tenant

#### Note

- In the configuration, you can define any number of property set types. However, you can create only one property set type per POST request. You can create a higher number of property set types for a package through multiple POST requests.
- For property set type of data categories *TimeSeriesData* and *DerivedData*, you can define property types of data type *GeoJSON*. However, you can create time series data or derived data for a property of type *GeoJSON* only via data ingestion from SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (Internet of Things Service).

#### → Recommendation

You can define up to 10 property set types for a package.

For property set type of data category *TimeSeriesData* or *DerivedData*, you can define up to 500 properties. However, for better performance, we highly recommend you to define not more than 200 properties per property set type.

### Naming convention for property set type and reference property set type

- Always prefixed with the package name.
- Can be a combination of alphanumeric characters and underscore.
- Each name must be fully qualified and contains a package name, colon as a separator and the object name. Maximum length of fully qualified name is 81 characters.
- Do not start the object name with a digit or an underscore.
- Length should be at least 3 characters.
- Camel case is generally allowed, however there is a uniqueness check carried out against textual case conflicts:
  - Two names with same characters of different textual case is not allowed. For example: A value "AbcDef" is not accepted when there is another value "abcDEF" that exists.
- Verify that a colon is used only to separate a package name and the object name. The syntax is: <package>:<name>. For example: core.automobiles:FlyWheel
- Name must be unique within a package.

### Naming convention for properties

- Only the following characters are supported:
  - Uppercase and lowercase alphabets *a* through *z* and *A* through *Z*
  - Numeric digits 0 through 9
  - Punctuation marks underscore (*\_*), hyphen (*-*), colon (*:*), and full-stop (*.*)

### Note

However, punctuation marks hyphen (-), colon (:), and full-stop (.) are not supported for data categories *TimeSeriesData* and *DerivedData*.

- Maximum length is 50 characters.
- Property names have to be unique for a given set of property set types.
- Property name must not start with a digit only for data categories *TimeSeriesData* and *DerivedData*.
- For properties that shall be used in rule processing scenarios, additional naming conventions are in effect. For more information, see [Naming Conventions for Rule Processing](#).

## Request

**URI:** `/ThingConfiguration/v1/Packages('<package name>')/PropertySetTypes`

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** `<thingconf>.c`

### Request Headers

Header	Required	Values
Accept-Language	No	Language of the property set type description. The default language is <i>en</i> .

### Request Parameters

Parameter	Required	Data Type	Description
package name	Yes	String	Unique name of the package

### Request Example

`/ThingConfiguration/v1/Packages('core.automobiles')/PropertySetTypes`

## Payload

**Format:** JSON

### Example 1

`/ThingConfiguration/v1/Packages('core.automobiles')/PropertySetTypes`

Creates a property set type `core.automobiles:Engine` in the package `core.automobiles`.

```
{
  "Name": "core.automobiles:Engine",
  "DataCategory": "TimeSeriesData",
  "Descriptions": [{
```



```

        "LanguageCode": "en",
        "Description": "Sensory parameters for Engine"
    }],
    "Properties": [{
        "Name": "Temperature",
        "Descriptions": [{
            "LanguageCode": "en",
            "Description": "Temperature of the Engine"
        }],
        "Type": "Numeric",
        "PropertyLength": "",
        "QualityCode": "1",
        "UnitOfMeasure": "AA"
    },
    {
        "Name": "Pressure",
        "Descriptions": [{
            "LanguageCode": "en",
            "Description": "Pressure of the Engine"
        }],
        "Type": "Numeric",
        "PropertyLength": "",
        "QualityCode": "0",
        "UnitOfMeasure": "BA"
    }
    ]
}

```

## Example 2

/ThingConfiguration/v1/Packages('core.automobiles')/PropertySetTypes

Creates a property set type `core.automobiles:Engine` in the package `core.automobiles`. The property `EngineTemperature` is referred to the property type `core.automobiles:Temperature`.

### Note

You can enable the property that is referred to a property type to support multiple languages.

```

{
    "Name": "core.automobiles:Engine",
    "DataCategory": "TimeSeriesData",
    "Descriptions": [{
        "LanguageCode": "en",
        "Description": "Sensory parameters for Engine"
    }],
    "Properties": [{
        "Name": "EngineTemperature",
        "Descriptions": [{
            "LanguageCode": "en",
            "Description": "Temperature of the Engine"
        }],
        "PropertyType": "core.automobiles:Temperature"
    },
    {
        "Name": "Pressure",
        "Descriptions": [{
            "LanguageCode": "en",
            "Description": "Pressure of the Engine"
        }],
        "Type": "Numeric",
        "PropertyLength": "",
        "QualityCode": "0",
        "UnitOfMeasure": "BA"
    }
    ]
}

```

```
}
```

### Example 3

```
/ThingConfiguration/v1/Packages('core.automobiles')/PropertySetTypes
```

Creates a reference property set type `core.automobiles:EngineReference` in the package `core.automobiles`.

```
{
  "Name": "core.automobiles:EngineReference",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Sensory parameters (reference) for Engine"
  }],
  "DataCategory": "ReferencePropertyData",
  "ReferredPropertySetType": "core.automobiles:Engine",
  "Properties": [{
    "Name": "TemperatureRefUoM",
    "AttributeType": "com.sap.iot.core.UoM",
    "ReferenceProperty": "Temperature"
  }, {
    "Name": "PressureRefLThreshold",
    "AttributeType": "com.sap.iot.core.LowerThreshold",
    "ReferenceProperty": "Pressure"
  }]
}
```

### Example 4

```
/ThingConfiguration/v1/Packages('core.automobiles')/PropertySetTypes
```

Creates a property set type `core.automobiles:BusEngine` copied from the already existing property set type `core.automobiles:Engine`. The method copies the entire structure of the source property set type to the new property set type. However, you can include the descriptions of the property set type in the `Descriptions` field along with the `CopySource` field. In which case, the method creates a new property set type with these descriptions instead of copying it from the source property set type.

```
{
  "Name": "core.automobiles:BusEngine",
  "CopySource": "core.automobiles:Engine"
}
```

### Note

Ensure that the source property set type belongs to a package with `scope=tenant` as you cannot create copy of a property set type that belongs to a package with `scope=private`.

### Example 5

```
/ThingConfiguration/v1/Packages('core.automobiles')/PropertySetTypes
```

Creates a property set type `core.automobiles:Car` in the package `core.automobiles` which supports multiple languages for the string type properties. In the following example, the string type properties **Color**, **Model**, and **LicenseNumber** support multiple languages by default as the property set type `core.automobiles:car` supports multiple languages. However, if you do not want the string type property **Model** to support multiple languages, you must add the field `IsMultiLingual = False` for the property.

```
{
  "Name": "core.automobiles:Style",
```

```

    "DataCategory": "MasterData",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Master data parameters for Car"
    }],
    "IsMultiLingual": "true",
    "Properties": [{
      "Name": "Color",
      "Descriptions": [{
        "LanguageCode": "en",
        "Description": "Color of the Car"
      }],
      "Type": "String",
      "PropertyLength": ""
    },
    {
      "Name": "Model",
      "Descriptions": [{
        "LanguageCode": "en",
        "Description": "Model of the Car"
      }],
      "Type": "String",
      "PropertyLength": "",
      "IsMultiLingual": "false"
    },
    {
      "Name": "ModelNumber",
      "Descriptions": [{
        "LanguageCode": "en",
        "Description": "Model of the Car"
      }],
      "Type": "Numeric",
      "PropertyLength": ""
    },
    {
      "Name": "LicenseNumber",
      "Descriptions": [{
        "LanguageCode": "en",
        "Description": "License number of the Car"
      }],
      "Type": "String",
      "PropertyLength": ""
    }
  ]
}

```

## Example 6

```
/ThingConfiguration/v1/Packages('core.automobiles')/PropertySetTypes
```

Creates a property set type `core.automobiles:Engine` in the package `core.automobiles` along with annotations for the properties. `core.automobiles:Pressure108` is a user-defined annotation and `com.sap.apptot.analytics:dimension` is an SAP delivered predefined annotation.

### Note

You can assign multiple unique annotations for a property. However, you cannot assign the same annotation more than once for a property.

```

{
  "Name": "core.automobiles:Engine",
  "DataCategory": "TimeSeriesData",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Sensory parameters for Engine"
  }

```

```

    }],
    "Properties": [{
      "Name": "Temperature",
      "Descriptions": [{
        "LanguageCode": "en",
        "Description": "Temperature of the Engine"
      }],
      "Type": "Numeric",
      "PropertyLength": "",
      "QualityCode": "1",
      "UnitOfMeasure": "AA",
      "Annotations": [{
        "Name": "core.automobiles:Pressure108"
      }]
    },
    {
      "Name": "Pressure",
      "Descriptions": [{
        "LanguageCode": "en",
        "Description": "Pressure of the Engine"
      }],
      "Type": "Numeric",
      "PropertyLength": "",
      "QualityCode": "0",
      "UnitOfMeasure": "BA",
      "Annotations": [{
        "Name": "com.sap.apptot.analytics:dimension"
      }]
    }
  ]
}

```

### Note

In the above example, it is not mandatory to provide the ADD operation for the annotation of the properties Temperature and Pressure. However, if you provide the ADD operation for the property Temperature, you must provide the ADD operation for the property Pressure as well.

## Response

### Response Status and Error Codes

Code	Description
201	Property set types created successfully for a specific package

### Response Payload

### Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* definition for this method has the structure as illustrated in the following examples.

The following is the response payload after creating the property set type `core.automobiles:Engine`.

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')",
      "type": "com.sap.apptot.PropertySetType"
    },
    "Name": "core.automobiles:Engine",
    "PackageName": "core.automobiles",
    "DataCategory": "TimeSeriesData",
    "Description": null,
    "ReferredPropertySetType": null,
    "NumberOfProperties": null,
    "IsMultiLingual": null,
    "Properties": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')/Properties"
      }
    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')/Descriptions"
      }
    },
    "Annotations": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')/Annotations"
      }
    }
  }
}
```

The following is the response payload after creating the reference property set type `core.automobiles:EngineReference`.

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngineReference')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngineReference')",
      "type": "com.sap.apptot.PropertySetType"
    },
    "Name": "core.automobiles:EngineReference",
    "PackageName": "core.automobiles",
    "DataCategory": "ReferencePropertyData",
    "Description": null,
    "ReferredPropertySetType": "core.automobiles:Engine",
    "NumberOfProperties": null,
    "IsMultiLingual": null,
    "Properties": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngineReference')/Properties"
      }
    }
  }
}
```

```

    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngineReference')/
Descriptions"
      }
    },
    "Annotations": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngineReference')/
Annotations"
      }
    }
  }
}

```

## Related Information

[Data Protection and Privacy \[page 322\]](#)

## 8.2.5.2 Read Details of a Property Set Type

With this method, you can read the details of the specified property set type.

### Request

**URI:** `/ThingConfiguration/v1/PropertySetTypes('<property set type name>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

### Request Headers

Header	Required	Values
Accept-Language	No	Language of the property set type description. The default language is <a href="#">en</a> .

## Request Parameters

Parameter	Required	Data Type	Description
property set type name	Yes	String	Name of the property set type

## Request Example

```
/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')
```

## Response

### Response Headers

Header	Description
Etag	Entity tag of the newly created property set type instance. The system generates a new value and replaces the old value every time a property set type is updated.

### Response Status and Error Codes

Code	Description
200	Details of the specified property set type retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The JSON for this method has the structure as depicted in the following examples:

### Example 1

```
(/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine'))
```

Retrieves details of the property set type `core.automobiles:Engine`

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')",
      "type": "com.sap.apptot.PropertySetType"
    },
    "Name": "core.automobiles:Engine",
    "PackageName": "core.automobiles",
    "DataCategory": "TimeSeriesData",
    "Description": "Sensory Parameters for Engine",
    "ReferredPropertySetType": null,
    "CopySource": null,
  }
}
```

```

    "NumberOfProperties": null,
    "IsMultiLingual": null,
    "Properties": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')/Properties"
      }
    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')/
Descriptions"
      }
    },
    "Annotations": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')/
Annotations"
      }
    }
  }
}

```

## Example 2

```

/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')?$format=json?
$expand=Description

```

Retrieves details of the property set type `core.automobiles:Engine` along with descriptions in all specified languages

```

{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/
ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/
ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')",
      "type": "com.sap.apptot.PropertySetType"
    },
    "Name": "core.automobiles:Engine",
    "PackageName": "core.automobiles",
    "DataCategory": "TimeSeriesData",
    "Description": null,
    "ReferredPropertySetType": null,
    "CopySource": null,
    "NumberOfProperties": null,
    "IsMultiLingual": null,
    "Properties": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')/Properties"
      }
    },
    "Descriptions": {
      "results": [{
        "LanguageCode": "en",
        "Description": "Sensory parameters for Engine"
      }]
    },
    "Annotations": {
      "__deferred": {

```



```

        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')/Annotations"
      }
    }
  }
}

```

### Example 3

```

/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')?$format=json?
$expand=Properties

```

Retrieves details of the property set type `core.automobiles:Engine` along with its properties

```

{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')",
      "type": "com.sap.apptot.PropertySetType"
    },
    "Name": "core.automobiles:Engine",
    "PackageName": "core.automobiles",
    "DataCategory": "TimeSeriesData",
    "Description": "Sensory parameters for Engine",
    "ReferredPropertySetType": null,
    "CopySource": null,
    "NumberOfProperties": null,
    "IsMultiLingual": null,
    "Properties": {
      "results": [{
        "Name": "variable_1",
        "PropertySetType": "core.automobiles:Engine",
        "Description": "Temperature of the Engine",
        "Type": "Numeric",
        "PropertyType": "",
        "PropertyLength": "",
        "QualityCode": "1",
        "UnitOfMeasure": null,
        "ReferenceProperty": null,
        "AttributeType": "",
        "Operation": "",
        "IsMultiLingual": null
      },
      {
        "Name": "variable_2",
        "PropertySetType": "core.automobiles:Engine",
        "Description": "Pressure of the Engine",
        "Type": "Numeric",
        "PropertyType": "",
        "PropertyLength": "",
        "QualityCode": "1",
        "UnitOfMeasure": null,
        "ReferenceProperty": null,
        "AttributeType": "",
        "Operation": "",
        "IsMultiLingual": null
      }
    ]
  },
  "Descriptions": {
    "__deferred": {

```

```

        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')/Descriptions"
      },
      "Annotations": {
        "__deferred": {
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')/Annotations"
        }
      }
    }
  }
}

```

#### Example 4

```

/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')?$format=json?
$expand=Properties/Annotations

```

Retrieves details of the property set type `core.automobiles:Engine` along with its properties and annotations defined for the properties

```

{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')",
      "type": "com.sap.apptot.PropertySetType"
    },
    "Name": "core.automobiles%3AEngine",
    "PackageName": "core.automobiles",
    "DataCategory": "TimeSeriesData",
    "Description": "Sensory parameters for Engine",
    "ReferredPropertySetType": null,
    "CopySource": null,
    "NumberOfProperties": null,
    "IsMultiLingual": null,
    "Properties": {
      "results": [{
        "__metadata": {
          "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Properties(Name='variable_1',PropertySetType='core.automobiles%3AEngine')",
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Properties(Name='variable_1',PropertySetType='core.automobiles%3AEngine')",
          "type": "com.sap.apptot.Property"
        },
        "Name": "Temperature",
        "PropertySetType": "core.automobiles:Engine",
        "Description": "Temperature of the Engine",
        "Type": "String",
        "PropertyLength": "",
        "QualityCode": "1",
        "UnitOfMeasure": null,
        "ReferenceProperty": "",
        "AttributeType": "",
        "Operation": "",
        "PropertyType": "",
        "IsMultiLingual": null,
        "Descriptions": {
          "__deferred": {

```

```

        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='variable_1',PropertySetType='core.automobiles%3AEngine')/
Descriptions"
    },
    "Annotations": {
        "results": [{
            "Name": "core.automobiles:Pressure108",
            "PackageName": "core.automobiles",
            "Description": null
        }]
    },
    "ReferenceProperties": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='variable_1',PropertySetType='core.automobiles%3AEngine')/
ReferenceProperties"
        }
    }
},
{
    "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles%3AEngine')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles%3AEngine')",
        "type": "com.sap.apptot.Property"
    },
    "Name": "Pressure",
    "PropertySetType": "'core.automobiles%3AEngine'",
    "Description": "",
    "Type": "Numeric",
    "PropertyLength": "",
    "QualityCode": "1",
    "UnitOfMeasure": "",
    "ReferenceProperty": "",
    "AttributeType": "",
    "Operation": "",
    "PropertyType": "",
    "IsMultiLingual": null,
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles%3AEngine')/
Descriptions"
        }
    },
    "Annotations": {
        "results": [{
            "Name": "com.sap.apptot.analytics:dimension",
            "PackageName": "core.automobiles",
            "Description": null
        }]
    },
    "ReferenceProperties": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles%3AEngine')/
ReferenceProperties"
        }
    }
}
}

```

```

    ],
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')/Descriptions"
      }
    },
    "Annotations": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')/Annotations"
      }
    }
  }
}

```

### 8.2.5.3 Read all Property Set Types

With this method, you can read all property set types.

This method retrieves all property set types to which the user has access authorization. It includes property set types from all packages from the same tenant.

In addition, you can retrieve a subset of all property set types, according to the filter criteria provided. For a collective request for a set of property set types, you have these options:

- You can restrict the set of property set types to be retrieved by filter criteria based on the property set type name, package name, data category, and description.
- You can define that the set of matching property set types shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ThingConfiguration/v1/PropertySetTypes`

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** `<thingconf>.r`

## Request Headers

Header	Required	Values
Accept-Language	No	Language of the property set type description. The default language is <a href="#">en</a> .
sap-iot-propertiescount	No	<p>Include a new field <code>NumberOfProperties</code> in the response payload</p> <p>Only if you specify the value <a href="#">true</a> for this parameter, the system includes the count of properties in a new field <code>NumberOfProperties</code> in the response payload. If you specify <a href="#">false</a> or do not use this header parameter, the system does not return the count of properties in the response payload.</p>

## Query Request Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$inlinecount	No	String	<p>Total number of property set types available; permissible values are <a href="#">allpages</a> and <a href="#">none</a>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"><li>• If you use this with the <code>\$top</code> and <code>\$skip</code> parameters, the result contains the total count of property set types available.</li><li>• If you use this with <code>\$filter</code> parameter, the result contains the total count of property set types based on the defined filter condition.</li></ul>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order.</p> <p>To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; only valid fields are property set type name, package name, data category, and description. The filter operators supported are <a href="#">eq</a>, <a href="#">lt</a>, <a href="#">gt</a>, <a href="#">le</a>, <a href="#">ge</a>, and <a href="#">ne</a>.</p> <p>Multiple filters are supported using the separators <a href="#">– and</a>, <a href="#">or</a>.</p> <p>You can use the string functions such as <a href="#">substringof</a>, <a href="#">startswith</a>, and <a href="#">endswith</a> to search data using <code>\$filter</code>. Valid fields are property set type name, package name, data category, and description.</p>

Parameter	Required	Data Type	Description
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are property set type name, package name, data category, description, and referred property set type name.

## Request Example

/ThingConfiguration/v1/PropertySetTypes

## Response

### Response Status and Error Codes

Code	Description
200	Details of all property set types retrieved successfully

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')",
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')",
        "type": "com.sap.apptot.PropertySetType"
      },
      "Name": "core.automobiles:Engine",
      "PackageName": "core.automobiles",
      "DataCategory": "TimeSeriesData",
      "Description": "Sensory parameters for Engine",
      "ReferredPropertySetType": "",
      "IsMultiLingual": null,
      "Properties": {
        "__deferred": {
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')/Properties"
        }
      },
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')/Descriptions"
        }
      }
    ]
  }
}
```

```

        "Annotations": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles%3AEngine')/Annotations"
            }
        },
        "__next": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/
ThingConfiguration/v1/PropertySetTypes?$format=json&$skiptoken=20"
    }
}

```

The following is an example payload with the count of properties included in the response payload:

```

{
    "d": {
        "results": [{
            "__metadata": {
                "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')",
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')",
                "type": "com.sap.apptot.PropertySetType"
            },
            "Name": "core.automobiles:Engine",
            "PackageName": "core.automobiles",
            "DataCategory": "TimeSeriesData",
            "Description": "Sensory parameters for Engine",
            "ReferredPropertySetType": "",
            "IsMultiLingual": null,
            "NumberOfProperties": 6,
            "Properties": {
                "__deferred": {
                    "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')/Properties"
                }
            },
            "Descriptions": {
                "__deferred": {
                    "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')/Descriptions"
                }
            },
            "Annotations": {
                "__deferred": {
                    "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')/Annotations"
                }
            }
        }],
        {
            "__metadata": {
                "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Car')",
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Car')",
                "type": "com.sap.apptot.PropertySetType"
            },
            "Name": "core.automobiles:Car",

```

```

        "PackageName": "core.automobiles",
        "DataCategory": "TimeSeriesData",
        "Description": "Sensory parameters for Car",
        "ReferredPropertySetType": "",
        "IsMultiLingual": null,
        "NumberOfProperties": 4,
        "Properties": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Car')/Properties"
            }
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Car')/Descriptions"
            }
        },
        "Annotations": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')/Annotations"
            }
        }
    }
}
]
}
}

```

## 8.2.5.4 Read all Property Set Types of a Package

With this method, you can read all property set types of the specified package.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

### Request

**URI:** `/ThingConfiguration/v1/Packages('<package name>')/PropertySetTypes`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`



## Request Headers

Header	Required	Values
Accept-Language	No	Language of the property set type description. The default language is <a href="#">en</a> .
sap-iot-propertiescount	No	<p>Include a new field <code>NumberOfProperties</code> in the response payload.</p> <p>Only if you specify the value <a href="#">true</a> for this parameter, the system includes the count of properties in a new field <code>NumberOfProperties</code> in the response payload. If you specify <a href="#">false</a> or do not use this header parameter, the system does not return the count of properties in the response payload.</p>

## Request Parameters

Parameter	Required	Data Type	Description
package name	Yes	String	Unique name of the package

## Query Request Parameters

Parameter	Required	Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$inlinecount	No	String	<p>Total number of property set types available; permissible values are <a href="#">allpages</a> and <a href="#">none</a>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"><li>• If you use this with the <code>\$top</code> and <code>\$skip</code> parameters, the result contains the total count of property set types available.</li><li>• If you use this with <code>\$filter</code> parameter, the result contains the total count of property set types based on the defined filter condition.</li></ul>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order.</p> <p>To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</p>

Parameter	Required	Type	Description
\$filter	No	String	<p>Filter condition to be applied; the only valid fields are property set type name, data category, and description. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the separators – <i>and</i>, <i>or</i>.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using \$filter. Valid fields are property set type name, data category, and description.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are property set type name, package name, data category, description, and referred property set type name.</p>

## Request Example

```
/ThingConfiguration/v1/Packages('<core.automobiles>')/PropertySetTypes
```

## Response

### Response Status and Error Codes

Code	Description
200	Details of all property set types retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')",
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertySetTypes('core.automobiles%3AEngine')",
        "type": "com.sap.apptot.PropertySetType"
      },
      "Name": "core.automobiles:Engine",
      "PackageName": "core.automobiles",
      "DataCategory": "TimeSeriesData",
    }
  ]
}
```

```

        "Description": "Sensory parameters for Engine",
        "ReferredPropertySetType": "",
        "IsMultiLingual": null,
        "Properties": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles%3AEngine')/Properties"
            }
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles%3AEngine')/Descriptions"
            }
        },
        "Annotations": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')/Annotations"
            }
        }
    }
}

```

The following is an example payload with the count of properties included in the response payload:

```

{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')",
        "type": "com.sap.apptot.PropertySetType"
      },
      "Name": "core.automobiles:Engine",
      "PackageName": "core.automobiles",
      "DataCategory": "TimeSeriesData",
      "Description": "Sensory parameters for Engine",
      "ReferredPropertySetType": "",
      "IsMultiLingual": null,
      "NumberOfProperties": 6,
      "Properties": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')/Properties"
        }
      },
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')/Descriptions"
        }
      },
      "Annotations": {
        "__deferred": {

```

```

        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')/Annotations"
    }
    },
    {
        "__metadata": {
            "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Style)",
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Style)",
            "type": "com.sap.apptot.PropertyType"
        },
        "Name": "core.automobiles:Style",
        "PackageName": "core.automobiles",
        "DataCategory": "MasterData",
        "Description": "Master data parameters for Car",
        "ReferredPropertySetType": "",
        "IsMultiLingual": null,
        "NumberOfProperties": 4,
        "Properties": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Style')/Properties"
            }
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Style')/Descriptions"
            }
        },
        "Annotations": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertySetTypes('core.automobiles:Engine')/Annotations"
            }
        }
    }
]
}
}

```

## 8.2.5.5 Update a Property Set Type

With this method, you can update the details of the specified property set type.

You can update the property set type as follows:

- You can assign annotation, delete or change the annotation assignment for a property set type
- You can add new properties, edit the existing properties, or delete the existing properties.
- You can assign or unassign the annotation for a property of a property set type.
- You can add the field `IsMultiLingual=true` to support multiple languages for its string type properties, if the property set type was not already supporting multiple languages. This is possible only if there is no data already existing for those properties.

- You can only increase the length of the property length, if data already exists for the property and the data type is **String**. For any other data types, you cannot change the property.
- You can add new properties by reusing the property types, if the following conditions are true:
  - Property type and property set type belongs to the same package
- You can change the data category of a property set type, irrespective of the data available for the properties. The only allowed possibility is *DerivedData* to *TimeSeriesData* and vice-versa.

The ETag value is mandatory to update a property set type. You must retrieve the ETag value of the property set type using the GET (/ThingConfiguration/v1/PropertySetTypes('<property set type name>')) service. The ETag value is assigned to the request header parameter If-Match in the PATCH request. If you do not include the ETag value in the request header or if the ETag value is not correct, the server displays an error message.

After updating a property set type, the system generates a new ETag value for the property set type and replaces the old value with the new value.

### Note

- You cannot change the name of the property set type.
- You cannot change the value of the IsMultiLingual filed for a property set type, if data already exists for its properties.
- The valid operations for the properties are ADD, DELETE, and UPDATE.
  - You cannot change the value of the field IsMultiLingual for a string type property, if data already exists for that property.
  - You can add the field IsMultiLingual=true Or IsMultiLingual=false for the string type property, if data does not exist and the property set type supports multiple languages.
  - You cannot delete the property, if data already exists for that property or if the property is referred in the reference property set type. You must first remove the reference by deleting the property in the reference property set type and then delete the property in the property set type.
  - You cannot change the type or the length of the property, if the property is referred in the reference property set type.
- The valid operations for the annotations of the property set type are ADD, DELETE, and UPDATE.
  - You can add annotation for a property set type if nothing is already assigned and even though the data exists for the properties
  - You cannot add multiple annotations for the property set type
  - You cannot add, delete, or change the annotation for a property set type if it is referred in another property set type
  - You cannot delete or change the already assigned annotation, if data exists for the properties of the property set type
- The valid operations for the annotations of the property are ADD and DELETE.
  - You can assign new annotations for a property even though the data exists for the property. It is not mandatory to specify the ADD operation field when you assign annotations for a property. However, if you want to provide the ADD operation, you must provide for all annotations of the property. If one or more annotations in the request payload is already assigned for the property and if you provide the ADD operation field, the service displays an error message.
  - You can unassign an annotation already assigned for a property by explicitly providing the DELETE operation in the request payload. If the property already contains data, you cannot delete the annotation.

- If you do not provide the operation field in the request payload, the service checks if the annotations provided in the request payload are already assigned for the property. The service assigns only the unassigned annotation(s) provided in the request payload for the property.
  - For example, assume annotations A1 and A2 are already assigned for the property Prop1. If you update the property Prop1 of the property set type with annotations A1 and A3, and without DELETE operation field, the service assigns annotation A3 for the property and the annotation A1 is ignored as it is already assigned.
  - You can assign or unassign multiple annotations for one or more properties in a single payload.
- If you are updating a reference property set type, you cannot change the referred property set type.

## Request

**URI:** `/ThingConfiguration/v1/PropertySetTypes('<property set type name>')`

**Operation Type:** CRUD

**HTTP Method:** *PATCH*

**Permissions:** `<thingconf>.u`

### Request Headers

Header	Required	Values
Accept-Language	No	Language of the property set type description. The default language is <i>en</i> .
If-Match	Yes	Latest ETag value of a specific property set type name that is updated using the <i>PATCH</i> request.

### Request Parameters

Parameter	Required	Data Type	Description
property set type name	Yes	String	Name of the property set type

### Request Example

`/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')`

## Payload

**Format:** JSON

Only media type JSON is supported. The JSON for this method has the structure as described in the following examples:

**Example 1:** The following example deletes the property `Color` for the property set type

`core.automobiles:Engine`.

```
{
  "DataCategory": "DerivedData",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Data for m4"
  }],
  "Properties": [
    {
      "Name": "Color",
      "Operation": "DELETE"
    }
  ]
}
```

**Example 2:** The following request payload assigns a new annotation for the property `Pressure` of the property set type `core.automobiles:Engine`.

```
{
  "DataCategory": "DerivedData",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Data for m4"
  }],
  "Properties": [
    {
      "Name": "Pressure",
      "Operation": "UPDATE",
      "Annotations": [{
        "Name": "core.automobiles:Pressure108",
        "Operation": "ADD",
      }]
    }
  ]
}
```

### Note

You cannot update the property set type name using the `PATCH` request. The property set type name specified in the request payload is ignored for update.

## Response

### Response Headers

Name	Description
Etag	Identifies the updated property set type

## Response Status and Error Codes

Code	Description
204	Details of the specified property set type updated successfully

## 8.2.5.6 Delete a Property Set Type

With this method, you can delete the specified property set type.

The [ETag](#) value is mandatory to delete a property set type. You must retrieve the [ETag](#) value of the property set type using the GET (/ThingConfiguration/v1/PropertySetTypes('<property set type name>')) service. The ETag value is assigned to the request header parameter If-Match in the [PATCH](#) request. If you do not include the ETag value in the request header or if the ETag value is not correct, the server displays an error message.

### Request

**URI:** /ThingConfiguration/v1/PropertySetTypes('<property set type name>')

**Operation Type:** CRUD

**HTTP Method:** [DELETE](#)

**Permissions:** <thingconf>.d

### Request Headers

Header	Required	Values
If-Match	Yes	Latest ETag value of a specific property set type that is updated using the <a href="#">PATCH</a> request.

### Request Parameters

Parameter	Required	Data Type	Description
property set type name	Yes	String	Unique name of the property set type

### Request Example

/ThingConfiguration/v1/PropertySetTypes('<property set type name>')



## Response

### Response Status and Error Codes

Code	Description
204	Property set type deleted successfully

## 8.2.5.7 Read all Properties of a Property Set Type

With this method, you can retrieve all properties that belong a property set type.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ThingConfiguration/v1/PropertySetTypes('<property set type name>')/Properties`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

### Request Headers

Header	Required	Values
<code>Accept-Language</code>	No	Language of the property set type description. The default language is en.

### Request Parameters

Parameter	Required	Data Type	Description
<code>property set type name</code>	Yes	String	Name of the property set type

### Query Request Parameters

Parameter	Required	Data Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set.
<code>\$skip</code>	No	Integer	Number of the first n records to be excluded from the result set.

Parameter	Required	Data Type	Description
\$inlinecount	No	String	<p>Total number of property set types available; permissible values are <a href="#">allpages</a> and <a href="#">none</a>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"> <li>• If you use this with the \$top and \$skip parameters, the result contains the total count of property set types available.</li> <li>• If you use this with \$filter parameter, the result contains the total count of property set types based on the defined filter condition.</li> </ul>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; only valid fields are property name, description, type, property length, quality code, and unit of measure.</p> <p>To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; only valid fields are property name, description, type, property length, quality code, and unit of measure. The filter operators supported are <a href="#">eq</a>, <a href="#">lt</a>, <a href="#">gt</a>, <a href="#">le</a>, <a href="#">ge</a>, and <a href="#">ne</a>.</p> <p>Multiple filters are supported using the separators – <a href="#">and</a>, <a href="#">or</a>.</p> <p>You can use the string functions such as <a href="#">substringof</a>, <a href="#">startswith</a>, and <a href="#">endswith</a> to search data using \$filter. Valid fields are property name, type, property set type.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; all fields in the response payload are supported.</p>

## Request Example

```
/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')/Properties
```

Retrieves all properties that belong to the property set type `core.automobiles:Engine`

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as depicted in the following examples:

### Example 1

/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')/Properties

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Temperature',PropertySetType='core.automobiles:Engine')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Temperature',PropertySetType='core.automobiles:Engine')",
        "type": "com.sap.apptot.Property"
      },
      "Name": "Temperature",
      "PropertySetType": "core.automobiles:Engine",
      "Description": "Temperature of the Engine",
      "Type": "Numeric",
      "PropertyLength": "",
      "QualityCode": "1",
      "UnitOfMeasure": "AA",
      "ReferenceProperty": null,
      "AttributeType": null,
      "Operation": null,
      "PropertyType": null,
      "IsMultilingual": null,
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Temperature',PropertySetType='core.automobiles:Engine')/
Descriptions"
        }
      },
      "Annotations": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Temperature',PropertySetType='core.automobiles:Engine')/
Annotations"
        }
      },
      "ReferenceProperties": {
        "__deferred": {
```

```

        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Temperature',PropertySetType='core.automobiles:Engine')/
ReferenceProperties"
    }
},
{
    "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles:Engine')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles:Engine')",
        "type": "com.sap.apptot.Property"
    },
    "Name": "Pressure",
    "PropertySetType": "core.automobiles:Engine",
    "Description": "Pressure of the Engine",
    "Type": "Numeric",
    "PropertyLength": null,
    "QualityCode": "0",
    "UnitOfMeasure": "BA",
    "ReferenceProperty": null,
    "AttributeType": null,
    "Operation": null,
    "PropertyType": null,
    "IsMultilingual": null,
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles:Engine')/
Descriptions"
        }
    },
    "Annotations": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles:Engine')/
Annotations"
        }
    },
    "ReferenceProperties": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles:Engine')/
ReferenceProperties"
        }
    }
}
]
}
}

```

## Example 2

/ThingConfiguration/v1/PropertySetTypes('core.automobiles:Engine')/Properties?  
\$expand=ReferenceProperties

```

{
    "d": {
        "results": [{
            "__metadata": {

```

```

        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Temperature',PropertySetType='core.automobiles:Engine')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Temperature',PropertySetType='core.automobiles:Engine')",
        "type": "com.sap.apptot.Property"
    },
    "Name": "Temperature",
    "PropertySetType": "core.automobiles:Engine",
    "Description": "Temperature of the Engine",
    "Type": "Numeric",
    "PropertyLength": null,
    "QualityCode": "1",
    "UnitOfMeasure": "AA",
    "ReferenceProperty": null,
    "AttributeType": null,
    "Operation": null,
    "PropertyType": null,
    "IsMultilingual": null,
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Temperature',PropertySetType='core.automobiles:Engine')/
Descriptions"
        }
    },
    "Annotations": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Temperature',PropertySetType='core.automobiles:Engine')/
Annotations"
        }
    },
    "ReferenceProperties": {
        "results": [{
            "Name": "TemperatureRefUoM",
            "PropertySetType": "core.automobiles:EngineReference",
            "Description": "Reference Temperature of the Engine",
            "Type": null,
            "PropertyLength": null,
            "QualityCode": null,
            "UnitOfMeasure": null,
            "ReferenceProperty": "Temperature",
            "AttributeType": "com.sap.iot.core.UoM",
            "Operation": null,
            "PropertyType": null,
            "IsMultilingual": null
        }]
    },
    {
        "__metadata": {
            "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles:Engine')",
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles:Engine')",
            "type": "com.sap.apptot.Property"
        },
        "Name": "Pressure",
        "PropertySetType": "core.automobiles:Engine",
        "Description": "Pressure of the Engine",
        "Type": "Numeric",
        "PropertyLength": null,

```

```

        "QualityCode": "0",
        "UnitOfMeasure": "BA",
        "ReferenceProperty": null,
        "AttributeType": null,
        "Operation": null,
        "PropertyType": null,
        "IsMultilingual": null,
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles:Engine')/
Descriptions"
            }
        },
        "Annotations": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='Pressure',PropertySetType='core.automobiles:Engine')/
Annotations"
            }
        },
        "ReferenceProperties": {
            "results": [{
                "Name": "PressureRefLThreshold",
                "PropertySetType": "core.automobiles:EngineReference",
                "Description": "Reference Pressure of the Engine",
                "Type": null,
                "PropertyLength": null,
                "QualityCode": null,
                "UnitOfMeasure": null,
                "ReferenceProperty": "Pressure",
                "AttributeType": "com.sap.iot.core.LowerThreshold",
                "Operation": null,
                "PropertyType": null,
                "IsMultilingual": null
            }]
        }
    }
}

```

### Example 3

```

/ThingConfiguration/v1/PropertySetTypes('core.automobiles:EngineReference')/
Properties?$expand=ReferenceProperties

```

```

{
    "d": {
        "results": [{
            "__metadata": {
                "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='TemperatureRefUoM',PropertySetType='core.automobiles:EngineRefer
ence')",
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='TemperatureRefUoM',PropertySetType='core.automobiles:EngineRefer
ence')",
                "type": "com.sap.apptiot.Property"
            },
            "Name": "TemperatureRefUoM",
            "PropertySetType": "core.automobiles:EngineReference",
            "Description": "Reference Temperature of the Engine",
            "Type": null,

```

```

        "PropertyLength": null,
        "QualityCode": null,
        "UnitOfMeasure": null,
        "ReferenceProperty": "Temperature",
        "AttributeType": "com.sap.iot.core.UoM",
        "Operation": null,
        "PropertyType": null,
        "IsMultilingual": null,
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='TemperatureRefUoM',PropertySetType='core.automobiles:EngineRefer
ence')/Descriptions"
            }
        },
        "Annotations": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='TemperatureRefUoM',PropertySetType='core.automobiles:EngineRefer
ence')/Annotations"
            }
        },
        "ReferenceProperties": {
            "results": []
        }
    },
    {
        "__metadata": {
            "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='PressureRefLThreshold',PropertySetType='core.automobiles:EngineR
eference')",
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='PressureRefLThreshold',PropertySetType='core.automobiles:EngineR
eference')",
            "type": "com.sap.appiot.Property"
        },
        "Name": "PressureRefLThreshold",
        "PropertySetType": "core.automobiles:EngineReference",
        "Description": "Reference Pressure of the Engine",
        "Type": null,
        "PropertyLength": null,
        "QualityCode": null,
        "UnitOfMeasure": null,
        "ReferenceProperty": "Pressure",
        "AttributeType": "com.sap.iot.core.LowerThreshold",
        "Operation": null,
        "PropertyType": null,
        "IsMultilingual": null,
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='PressureRefLThreshold',PropertySetType='core.automobiles:EngineR
eference')/Descriptions"
            }
        },
        "Annotations": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Properties(Name='PressureRefLThreshold',PropertySetType='core.automobiles:EngineR
eference')/Annotations"
            }
        }
    },

```

```

        "ReferenceProperties": {
            "results": []
        }
    }
}

```

## 8.2.6 Property Type

The property type defines the properties of a thing and they have a specific data type assigned. You can group the properties under a property set type. In addition, you can refer property types to define properties while creating a property set type. For example, the color and model of the car, pressure and rotation speed of the wheel, or capacity and transmission of an engine are some of the property types.

**Resource Path:** `http://<server address>[:<port number>][<path>]/ThingConfiguration`

**Example for a base URI in a cloud foundry environment:** `https://config-thing-sap.cfapps.eu10.hana.ondemand.com/ThingConfiguration`

### Operations

#### CRUD Operations

HTTP Method	Operation	URI	Scope
<i>POST</i>	<a href="#">Create a Property Type [page 479]</a>	<code>/v1/Packages('&lt;package name&gt;')/PropertyTypes</code>	<code>&lt;thingconf&gt;.c</code>
<i>GET</i>	<a href="#">Read Details of a Property Type [page 483]</a>	<code>/v1/PropertyTypes('&lt;property type name&gt;')</code>	<code>&lt;thingconf&gt;.r</code>
<i>GET</i>	<a href="#">Read all Property Types [page 489]</a>	<code>/v1/PropertyTypes</code>	<code>&lt;thingconf&gt;.r</code>
<i>GET</i>	<a href="#">Read all Property Types of a Package [page 493]</a>	<code>/v1/Packages('package name')/PropertyTypes</code>	<code>&lt;thingconf&gt;.r</code>
<i>PATCH</i>	<a href="#">Update a Property Type [page 497]</a>	<code>/v1/PropertyTypes('&lt;property type name&gt;')</code>	<code>&lt;thingconf&gt;.u</code>
<i>DELETE</i>	<a href="#">Delete a Property Type [page 500]</a>	<code>/v1/PropertyTypes('&lt;property type name&gt;')</code>	<code>&lt;thingconf&gt;.d</code>

#### Property Type Object Properties

Property	Type	Mandatory	Maximum Length	Description
Name	String	Yes	81	Name of the property type and it must be unique within a package



Property	Type	Mandatory	Maximum Length	Description
Description	String	No	60	Description of the property type for each language with the corresponding ISO language code
Type	enum	Yes	30	Type used by the property type
PropertyLength	string	Yes (For more information on supported data types, refer to <a href="#">Value Data Types for Property Types and Properties</a> [page 1229])		Storage length for the property type specified as an integer value

#### i Note

Data type *LargeString* is applicable only for data categories *TimeSeriesData* and *DerivedData*.

Property	Type	Mandatory	Maximum Length	Description
QualityCode	integer	No	1	<p>Indicates if the property type has a quality code attached. A value 0 implies there is no quality code attached; value 1 implies that the property type has a quality code available.</p> <p>Quality code is supported only for property set type of data category <i>TimeSeriesData</i> and <i>DerivedData</i>. You can only attach a quality code for a property type of data type <i>Numeric</i>, <i>NumericFlexible</i>, and <i>String</i>.</p> <div> <p><b>i Note</b></p> <p>Only if the property type has a quality code attached, an additional property <code>&lt;property_name&gt;_QC</code> is supported for storing the quality code value.</p> </div>
UnitOfMeasure	string	No	5	Indicates the unit of measure in which the property type is measured

Property	Type	Mandatory	Maximum Length	Description
Values	Enum	No	254	<p>Pre-defined values that can be assigned for the properties defined using the property type. If you specify enumerated values in this field, you can assign only these pre-defined values for the properties while creating data for a thing. If you do not specify any value in this field, you can define your own values while creating data for a thing.</p> <p>In addition, you can add description for each of the enumeration values in each language with the corresponding ISO language code.</p> <p>Data type supported are <i>String</i>, <i>Numeric</i>, <i>NumericFlexible</i>, and <i>NumericDecimal</i>.</p>

### Property Type Descriptions

Property	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the object description
Description	String	255	Description of the object

## 8.2.6.1 Create a Property Type

With this method, you can create a property type in the specified package.

Based on the scope of the package, the property type can be reused as follows:

- Only in the property set types within a package, if the scope of the package is *private*
- Only in the property set types within a tenant, if the scope of the package is *tenant*

## Naming convention

- Always prefixed with the package name.
- Can be a combination of alphanumeric characters and underscore.
- Each name must be fully qualified and contains a package name, colon as a separator and the object name. Maximum length of fully qualified name is 81 characters.
- Do not start the object name with a digit or an underscore.
- Length should be at least 3 characters.
- Verify that a colon is used only to separate a package name and the object name. The syntax is: `<package>:<name>`. For example: `core.automobiles:FlyWheel`
- Name must be unique within a package. However, two property names with same characters of different textual case is allowed. For example: A value "AbcDef" is accepted when there is another value "abcDEF" that exists.

## Request

**URI:** `/ThingConfiguration/v1/Packages('<package name>')/PropertyTypes`

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** `<thingconf>.c`

### Request Parameters

Parameter	Required	Data Type	Description
package name	Yes	String	Unique name of the package

### Request Example

`/ThingConfiguration/v1/Packages('core.automobiles')/PropertyTypes`

## Payload

**Format:** *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples.

The following is a request payload to create a property type `core.automobiles:Temperature` without enumeration values. In such scenario, you can assign any values for the properties while creating data for a thing instance.

```
{
  "Name": "core.automobiles:Temperature",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Temperature"
  }]
}
```

```

    }},
    "Type": "String",
    "PropertyLength": "128",
    "UnitOfMeasure": "C",
    "QualityCode": "1"
  }
}

```

The following is a request payload to create a property type `core.automobiles:Temperature` with enumeration values **High**, **Medium**, and **Low** defined for the property. You can assign only these values (High, Medium, and Low) for the property while creating data for a thing.

```

{
  "Name": "core.automobiles:Temperature",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Temperature"
  }],
  "Type": "String",
  "PropertyLength": "128",
  "Values": [
    { "Value": "High" },
    { "Value": "Medium" },
    { "Value": "Low" }
  ],
  "UnitOfMeasure": "C",
  "QualityCode": "1"
}

```

The following is a request payload to create a property type `core.automobiles:Temperature` with enumeration values **High**, **Medium**, and **Low** along with their descriptions defined for the property. You can assign only these values (High, Medium, and Low) for the property while creating data for a thing.

```

{
  "Name": "core.automobiles:Temperature",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Temperature"
  }],
  "Type": "String",
  "PropertyLength": "128",
  "Values": [{
    "Value": "High",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "High temperature"
    },
    {
      "LanguageCode": "de",
      "Description": "hohe Temperatur"
    },
    {
      "LanguageCode": "fr",
      "Description": "haute température"
    },
    {
      "LanguageCode": "ch",
      "Description": "高温"
    }
  ]
}, {
  "Value": "Medium",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "medium temperature"
  }],
  "UnitOfMeasure": "C",
  "QualityCode": "1"
}

```

```

    {
      "LanguageCode": "de",
      "Description": "Mittlere temperatur"
    },
    {
      "LanguageCode": "fr",
      "Description": "Température moyenne"
    },
    {
      "LanguageCode": "ch",
      "Description": "中等温度"
    }
  ]
}, {
  "Value": "Low",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Low temperature"
  },
  {
    "LanguageCode": "de",
    "Description": "niedrige Temperatur"
  },
  {
    "LanguageCode": "fr",
    "Description": "basse température"
  },
  {
    "LanguageCode": "ch",
    "Description": "低温"
  }
  ]
},
{
  "UnitOfMeasure": "C",
  "QualityCode": "1"
}

```

## Response

### Response Status and Error Codes

Code	Description
201	Property type created successfully for a specific package

## Payload

**Format:** [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as depicted in the following example.

The following is the response payload after creating the property type `core.automobiles:Temperature` in the package `core.automobiles`:

```
{
```

```

    "d": {
      "__metadata": {
        "id": "https://config-thing-sap-ci.cfapps.staging.sic.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
        "uri": "https://config-thing-sap-ci.cfapps.staging.sic.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
        "type": "com.sap.apptot.PropertyType"
      },
      "Name": "core.automobiles:Temperature",
      "Description": "Temperature",
      "Type": "String",
      "PropertyLength": "128",
      "QualityCode": "1",
      "UnitOfMeasure": "C",
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-thing-sap-ci.cfapps.staging.sic.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')/Descriptions"
        }
      },
      "Values": {
        "__deferred": {
          "uri": "https://config-thing-sap-ci.cfapps.staging.sic.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')/Values"
        }
      }
    }
  }
}

```

## Related Information

[Property Type \[page 476\]](#)

### 8.2.6.2 Read Details of a Property Type

With this method, you can retrieve the details of a specific property type.

#### Request

**URI:** /ThingConfiguration/v1/PropertyTypes('<property type name>')

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <thingconf>.r

## Request Parameters

Parameter	Required	Data Type	Description
property type name	Yes	String	Name of the property type

## Request Example

```
/ThingConfiguration/v1/PropertyTypes('core.automobiles:Temperature')
```

## Response

### Response Headers

Header	Description
Etag	Entity tag of the newly created property type instance. The system generates a new value and replaces the old value every time a property type is updated.

### Response Status and Error Codes

Code	Description
200	Details of a specific property type retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as depicted in the following examples

### Example 1

```
/ThingConfiguration/v1/PropertyTypes('core.automobiles:Temperature')
```

Retrieves the details of the property type `core.automobiles:Temperature`

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
      "type": "com.sap.apptot.PropertyType"
    },
    "Name": "core.automobiles:Temperature",
    "Description": "Temperature",
    "Type": "String",
    "PropertyLength": "128",
    "QualityCode": "1",
    "UnitOfMeasure": "C",
  }
}
```



```

    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')/Descriptions"
      }
    },
    "Values": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')/Values"
      }
    }
  }
}

```

## Example 2

```

/ThingConfiguration/v1/PropertyTypes('core.automobiles:Temperature')?
$expand=Descriptions

```

Retrieves details of the property type `core.automobiles:Temperature` along with the description of the property type expanded

```

{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
      "type": "com.sap.apptot.PropertyType"
    },
    "Name": "core.automobiles:Temperature",
    "Description": null,
    "Type": "String",
    "PropertyLength": "128",
    "QualityCode": "1",
    "UnitOfMeasure": "C",
    "Descriptions": {
      "results": [
        {
          "LanguageCode": "en",
          "Description": "Temperature"
        }
      ]
    },
    "Values": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles:Temperature')/Values"
      }
    }
  }
}

```

## Example 3

```

/ThingConfiguration/v1/PropertyTypes('core.automobiles:Temperature')?$expand=Values

```

Retrieves details of the property type `core.automobiles:Temperature` along with the values of the property type expanded

```

{
  "d": {

```

```

    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/
ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/
ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
      "type": "com.sap.apptot.PropertyType"
    },
    "Name": "core.automobiles:Temperature",
    "Description": "Temperature",
    "Type": "String",
    "PropertyLength": "128",
    "QualityCode": "1",
    "UnitOfMeasure": "C",
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')/
Descriptions"
      }
    },
    "Values": {
      "results": [
        {
          "Value": "High",
          "Description": "High temperature"
        },
        {
          "Value": "Medium",
          "Description": "Medium temperature"
        },
        {
          "Value": "Low",
          "Description": "Low temperature"
        }
      ]
    }
  }
}

```

#### Example 4

/ThingConfiguration/v1/PropertyTypes('core.automobiles:Temperature')?\$expand=Values

Retrieves details of the property type `core.automobiles:Temperature` along with the values of the property type expanded. If there are no descriptions defined for the enumeration values, the method assigns the value *null* for the descriptions as shown in the below example:

```

{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/
ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/
ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
      "type": "com.sap.apptot.PropertyType"
    },
    "Name": "core.automobiles:Temperature",
    "Description": "Temperature",
    "Type": "String",
    "PropertyLength": "128",
    "QualityCode": "1",
    "UnitOfMeasure": "C",
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')/
Descriptions"
      }
    }
  }
}

```

```

    }
  },
  "Values": {
    "results": [
      {
        "Value": "High",
        "Description": null
      },
      {
        "Value": "Medium",
        "Description": null
      },
      {
        "Value": "Low",
        "Description": null
      }
    ]
  }
}

```

### Example 5

```

/ThingConfiguration/v1/PropertyTypes('core.automobiles:Temperature')?
$expand=Values/Descriptions

```

Retrieves details of the property type `core.automobiles:Temperature` along with the descriptions of the enumeration values expanded

```

{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
      "type": "com.sap.apptot.PropertyType"
    },
    "Name": "core.automobiles:Temperature",
    "Description": "Temperature",
    "Type": "String",
    "PropertyLength": "128",
    "QualityCode": "1",
    "UnitOfMeasure": "C",
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')/Descriptions"
      }
    },
    "Values": {
      "results": [
        {
          "__metadata": {
            "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Values('High')",
            "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Values('High')",
            "type": "com.sap.apptot.Value"
          },
          "Value": "High",
          "Description": null,
          "Descriptions": {
            "results": [
              {
                "LanguageCode": "en",

```

```

        "Description": "High temperature"
      },
      {
        "LanguageCode": "de",
        "Description": "hohe Temperatur"
      },
      {
        "LanguageCode": "fr",
        "Description": "haute température"
      },
      {
        "LanguageCode": "ch",
        "Description": "高温"
      }
    ]
  },
  {
    "__metadata": {
      "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Values('Medium')",
      "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Values('Medium')",
      "type": "com.sap.apptot.Value"
    },
    "Value": "Medium",
    "Description": null,
    "Descriptions": {
      "results": [
        {
          "LanguageCode": "en",
          "Description": "Medium temperature"
        },
        {
          "LanguageCode": "de",
          "Description": "Mittlere temperatur"
        },
        {
          "LanguageCode": "fr",
          "Description": "Température moyenne"
        },
        {
          "LanguageCode": "ch",
          "Description": "中等温度"
        }
      ]
    }
  },
  {
    "__metadata": {
      "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Values('Low')",
      "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Values('Low')",
      "type": "com.sap.apptot.Value"
    },
    "Value": "Low",
    "Description": null,
    "Descriptions": {
      "results": [
        {
          "LanguageCode": "en",
          "Description": "Low temperature"
        },
        {
          "LanguageCode": "de",
          "Description": "niedrige Temperatur"
        }
      ]
    }
  }
]

```

```

    {
      "LanguageCode": "fr",
      "Description": "basse température"
    },
    {
      "LanguageCode": "ch",
      "Description": "低温"
    }
  ]
}

```

## Related Information

[Property Type \[page 476\]](#)

### 8.2.6.3 Read all Property Types

With this method, you can retrieve all property types.

This method retrieves all property types to which the user has access authorization. It includes property types from all packages from the same tenant.

In addition, you can retrieve a subset of all property types, according to the filter criteria provided. For a collective request for a set of property types, you have these options:

- You can restrict the set of property types to be retrieved by filter criteria based on the property type name, description, type, property length, quality code, and unit of measure.
- You can define that the set of matching property types shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ThingConfiguration/v1/PropertyTypes`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

## Request Parameters

None

### Query Request Parameters

Parameter	Required	Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$inlinecount	No	String	<p>Total number of property types available; permissible values are <a href="#">allpages</a> and <a href="#">none</a>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"><li>• If you use this with the \$top and \$skip parameters, the result contains the total count of thing types available.</li><li>• If you use this with \$filter parameter, the result contains the total count of thing types based on the defined filter condition.</li></ul>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; valid fields are name, description, type, property length, quality code, and unit of measure.</p> <p>To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</p>

Parameter	Required	Type	Description
\$filter	No	String	<p>Filter condition to be applied; the valid fields are name, description, type, property length, quality code, and unit of measure. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the separators – <i>and</i>, <i>or</i>.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using <i>\$filter</i>.</p>
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are name, description, type, property length, quality code, and unit of measure.

## Request Example

/ThingConfiguration/v1/PropertyTypes

## Response

### Response Status and Error Codes

Code	Description
200	Details of all property types retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

```
{
  "d": {
    "results": [{
      "__metadata": {
```

```

        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3ATemperature')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3ATemperature')",
        "type": "com.sap.apptot.PropertyType"
    },
    "Name": "core.automobiles:Temperature",
    "Description": "Temperature",
    "Type": "String",
    "PropertyLength": "128",
    "QualityCode": "1",
    "UnitOfMeasure": "C",
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3ATemperature')/Descriptions"
        }
    },
    "Values": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3ATemperature')/Values"
        }
    }
},
{
    "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3APressure')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3APressure')",
        "type": "com.sap.apptot.PropertyType"
    },
    "Name": "core.automobiles:Pressure",
    "Description": "Pressure",
    "Type": "String",
    "PropertyLength": "128",
    "QualityCode": "1",
    "UnitOfMeasure": "AA",
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3APressure')/Descriptions"
        }
    },
    "Values": {
        "__deferred": {
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3APressure')/Values"
        }
    }
}
]
}
}

```



## Related Information

[Property Type \[page 476\]](#)

### 8.2.6.4 Read all Property Types of a Package

With this method, you can retrieve all property types for the specified package.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ThingConfiguration/v1/Packages('package name')/PropertyTypes`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

## Request Parameters

Parameter	Required	Data Type	Description
<code>package name</code>	Yes	String	Unique name of the package

## Query Request Parameters

Parameter	Required	Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set.
<code>\$skip</code>	No	Integer	Number of the first n records to be excluded from the result set.

Parameter	Required	Type	Description
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; valid fields are name, description, type, property length, quality code, and unit of measure.</p> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are name, description, type, property length, quality code, and unit of measure.</p>
\$inlinecount	No	String	<p>Total number of property types available; permissible values are <i>allpages</i> and <i>none</i>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"> <li>• If you use this with the <i>\$top</i> and <i>\$skip</i> parameters, the result contains the total count of property types available.</li> <li>• If you use this with <i>\$filter</i> parameter, the result contains the total count of property types based on the defined filter condition.</li> </ul>

Parameter	Required	Type	Description
\$filter	No	String	<p>Filter condition to be applied; valid fields are name, description, type, property length, quality code, and unit of measure. The filter operators supported are <a href="#">eq</a>, <a href="#">lt</a>, <a href="#">gt</a>, <a href="#">le</a>, <a href="#">ge</a>, and <a href="#">ne</a>.</p> <p>Multiple filters are supported using the separators – <a href="#">and</a>, <a href="#">or</a>.</p> <p>You can use the string functions such as <a href="#">substringof</a>, <a href="#">startswith</a>, and <a href="#">endswith</a> to search data using \$filter. Only valid field is name.</p>

## Request Example

```
/ThingConfiguration/v1/Packages('core.automobiles')/PropertyTypes
```

## Response

### Response Status and Error Codes

Code	Description
200	Details of property type of a specific package are retrieved successfully

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
          "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/PropertyTypes('core.automobiles%3ATemperature')",
          "type": "com.sap.apptot.PropertyType"
        },

```

```

        "Name": "core.automobiles:Temperature",
        "Description": "Temperature",
        "Type": "String",
        "PropertyLength": "",
        "QualityCode": "1",
        "UnitOfMeasure": "C",
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3ATemperature')/Descriptions"
            }
        },
        "Values": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3ATemperature')/Values"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3APressure')",
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3APressure')",
            "type": "com.sap.apptot.PropertyType"
        },
        "Name": "core.automobiles:Pressure",
        "Description": "Pressure",
        "Type": "String",
        "PropertyLength": "",
        "QualityCode": "1",
        "UnitOfMeasure": "C",
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3APressure')/Descriptions"
            }
        },
        "Values": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3APressure')/Values"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3AColor')",
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3AColor')",
            "type": "com.sap.apptot.PropertyType"
        },
        "Name": "core.automobiles:Color",
        "Description": "Color",
        "Type": "String",
        "PropertyLength": "128",
        "QualityCode": "1",
        "UnitOfMeasure": "C",

```

```

        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3AColor')/Descriptions"
            }
        },
        "Values": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3AColor')/Values"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3AFinish')",
            "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3AFinish')",
            "type": "com.sap.apptot.PropertyType"
        },
        "Name": "core.automobiles:Finish",
        "Description": "Color",
        "Type": "String",
        "PropertyLength": "128",
        "QualityCode": "1",
        "UnitOfMeasure": "C",
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3AFinish')/Descriptions"
            }
        },
        "Values": {
            "__deferred": {
                "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
PropertyTypes('core.automobiles%3AFinish')/Values"
            }
        }
    }
]
}
}

```

## Related Information

[Property Type \[page 476\]](#)

### 8.2.6.5 Update a Property Type

With this method, you can update a property type.

The `ETag` value is mandatory to update a property type. You must retrieve the `ETag` value of the thing type using the `GET (/ThingConfiguration/v1/PropertyTypes('<property type name>'))` service. The

ETag value is assigned to the request header parameter `If-Match` in the `PATCH` request. If you do not include the ETag value in the request header or if the ETag value is not correct, the server displays an error message.

- You can add, edit, or delete the following for a property type:
  - Data type
  - Length
  - Descriptions
  - Enumeration values
  - Description of enumeration values
  - Unit of measure
  - Quality code
- You cannot change the name of the property type using `PATCH` request.

#### Note

- You cannot update a property type if it is already reused in a property set type.
- After updating a property type, the system generates a new ETag value for the thing type and replaces the old value with the new value.

## Request

**URI:** `/ThingConfiguration/v1/PropertyTypes('<property type name>')`

**Operation Type:** CRUD

**HTTP Method:** `PATCH`

**Permissions:** `<thingconf>.u`

### Request Header Parameters

Header	Required	Values
If-Match	Yes	Latest ETag value of a specific property type that is updated using the <code>PATCH</code> request.

### Request Parameters

Parameter	Required	Data Type	Description
property type name	Yes	String	Unique name of the property type.

### Request Example

`/ThingConfiguration/v1/PropertyTypes('core.automobiles:Temperature')`

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

The following request payload is used to add new enumeration values `High` and `Medium` along with their descriptions for the property type `core.automobiles:Temperature`

```
{
  "Name": "core.automobiles:Temperature",
  "Type": "String",
  "Values": [{
    "Value": "High",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "High temperature"
    }]
  }, {
    "Value": "Medium",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Medium temperature"
    }]
  }],
  "UnitOfMeasure": "C",
  "QualityCode": "1"
}
```

## Response

### Response Headers

Name	Description
Etag	Identifies the updated thing type

### Response Status and Error Codes

Code	Description
204	Property type updated successfully

## Related Information

[Property Type \[page 476\]](#)

## 8.2.6.6 Delete a Property Type

With this method, you can delete the specified property type.

The ETag value is mandatory to delete a property type. You must retrieve the ETag value of the property type using the `GET(/ThingConfiguration/v1/PropertyTypes('<property type name>'))` service. The ETag value is assigned to the request header parameter `If-Match` in the `DELETE` request. If you do not include the ETag value in the request header or if the ETag value is not correct, the server displays an error message.

### Note

You cannot delete a property type if it is already reused in a property set type.

## Request

**URI:** `/ThingConfiguration/v1/PropertyTypes('<property type name>')`

**Operation Type:** CRUD

**HTTP Method:** `DELETE`

**Permissions:** `<thingconf>.d`

### Request Header Parameters

Header	Required	Values
If-Match	Yes	Latest ETag value of a specific property type that is updated using the <code>DELETE</code> request.

### Request Parameters

Parameter	Required	Data Type	Description
property type name	Yes	String	Unique name of the property type

### Request Example

`/ThingConfiguration/v1/PropertyTypes('core.automobiles:Temperature')`



## Response

### Response Status and Error Codes

Code	Description
204	Property type deleted successfully

## Related Information

[Property Type \[page 476\]](#)

## 8.2.7 Annotation

Annotation is an array of attributes used to qualify properties of a property set type. You can define annotations in a package and assign it only to property set types that belong to the same package. In addition to the annotations you defined, you can also assign the predefined annotations offered by SAP.

The following is the only supported predefined annotation for a property type:

- `com.sap.appiot.analytics:dimension`
  - You can assign this annotation only to the property types of the following predefined value data types:
    - String
    - ThingID
    - BPID
  - However, if you have assigned quality code to the property type of the predefined value data types listed above, you cannot assign the predefined annotation `com.sap.appiot.analytics:dimension`.

The following are the supported predefined annotations for a property set type:

- `com.sap.appiot.security:pii`
  - You can assign this annotation to a property set type that contains data to identify a person or subject. Data such as nick name, birthdate, and email address are few examples that can be categorized as data that identifies a person.
- `com.sap.appiot.security:spi`
  - You can assign this annotation to a property set type that contains critical data of a person. Data such as identification number, ethnic origin, and biometric data are few examples that can be categorized as critical data of a person.

### i Note

- You cannot assign the predefined annotations for property set types of data category **ReferencePropertyData**. If the referred property set type is considered as personal information or sensitive personal information by assigning the predefined annotations (`com.sap.appiot.security:pii` and `com.sap.appiot.security:spi`), the reference property

data such as threshold values, unit of measure, and target values are also treated as personal information or sensitive personal information.

- You can assign multiple unique annotations for a property. However, you cannot assign the same annotation more than once for a property.

**Resource Path:** `http://<server address>[:<port number>][<path>]/ThingConfiguration`

**Example for a base URI in a cloud foundry environment:** `https://config-thing-sap.cfapps.eu10.hana.ondemand.com/ThingConfiguration`

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scope
<i>POST</i>	<a href="#">Create Annotation [page 503]</a>	<code>/v1/Packages ('&lt;package name&gt;') / Annotations</code>	<code>&lt;thingconf&gt;.c</code>
<i>GET</i>	<a href="#">Read Details of an Annotation [page 505]</a>	<code>/v1/Annotations ('&lt;annotation name&gt;')</code>	<code>&lt;thingconf&gt;.r</code>
<i>GET</i>	<a href="#">Read all Annotations of a Package [page 507]</a>	<code>/v1/Package ('&lt;package name&gt;') / Annotations</code>	<code>&lt;thingconf&gt;.r</code>
<i>PATCH</i>	<a href="#">Update an Annotation [page 510]</a>	<code>/v1/ Annotations ('&lt;annotation name&gt;')</code>	<code>&lt;thingconf&gt;.u</code>
<i>DELETE</i>	<a href="#">Delete an Annotation [page 511]</a>	<code>/v1/Annotations ('&lt;annotation name&gt;')</code>	<code>&lt;thingconf&gt;.d</code>

### Annotation Object Properties

Property	Data Type	Mandatory	Description
Name	String	Yes	Name of the annotation
LanguageCode	String	No	Indicates the ISO language code in which the annotation is described
Description	String	No	Description of the annotation

## Related Information

[Data Protection and Privacy \[page 322\]](#)

## 8.2.7.1 Create Annotation

With this method, you can create annotation in the specified package.

### i Note

In the configuration, you can define any number of annotations. However, you can create only one annotation per `POST` request. You can create more number of annotations for a package through multiple `POST` requests.

### → Recommendation

You can define up to 10 annotations for a package.

### Naming convention

- Always prefixed with the package name.
- Can be a combination of alphanumeric characters and underscore.
- Each name must be fully qualified and contains a package name, colon as a separator and the object name. Maximum length of fully qualified name is 81 characters.
- Do not start the object name with a digit or an underscore.
- Length should be at least 3 characters.
- Camel case is generally allowed, however there is a uniqueness check carried out against textual case conflicts:
  - Two names with same characters of different textual case is not allowed. For example: A value "AbcDef" is not accepted when there is another value "abcDEF" that exists.
- Verify that a colon is used only to separate a package name and the object name. The syntax is: `<package>:<name>`. For example: `core.automobiles:FlyWheel`
- Name must be unique within a package.

### Request

**URI:** `/ThingConfiguration/v1/Packages('<package name>')/Annotations`

**Operation Type:** CRUD

**HTTP Method:** `POST`

**Permissions:** `<thingconf>.c`

### Request Parameters

Parameter	Required	Data Type	Description
package name	Yes	String	Unique name of the package

### Request Example

**Format:** `JSON`

/ThingConfiguration/v1/Packages('core.automobiles')/Annotations

```
{
  "Name": "core.automobiles:Pressure108",
  "Descriptions": [
    {
      "LanguageCode": "en",
      "Description": "Description of annotation in en"
    },
    {
      "LanguageCode": "de",
      "Description": "Description of annotation in de"
    }
  ]
}
```

## Response

### Response Status and Error Codes

Code	Description
201	Annotations created successfully.

### Response Payload

### Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Annotations('core.automobiles%3APressure108')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Annotations('core.automobiles%3APressure108')",
      "type": "com.sap.apiot.Annotation"
    },
    "Name": "core.automobiles:Pressure108",
    "PackageName": "core.automobiles",
    "Description": "Description of annotation in en",
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Annotations('core.automobiles%3APressure108')/Descriptions"
      }
    }
  }
}
```

## 8.2.7.2 Read Details of an Annotation

With this method, you can retrieve details of the specified annotation.

### Request

**URI:** `/ThingConfiguration/v1/Annotations('<annotation name>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

### Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	String	Name of the annotation

### Request Example

```
/ThingConfiguration/v1/Annotations('core.automobiles:Pressure108')
```

### Response

#### Response Headers

Header	Description
Etag	Entity tag of the newly created annotation instance. The system generates a new value and replaces the old value every time the annotation is updated.

#### Response Status and Error Codes

Code	Description
200	Annotation retrieved successfully

### Payload

**Format:** [JSON](#)

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as depicted in the following examples:

Request without the query parameter `$expand(/ThingConfiguration/v1/Annotations('core.automobiles:Pressure108'))` returns the following response:

```
{
  "d": {
    "_metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Annotations('core.automobiles:Pressure108')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Annotations('core.automobiles:Pressure108')",
      "type": "com.sap.apptot.Annotation"
    },
    "Name": "core.automobiles:Pressure108",
    "PackageName": "core.automobiles",
    "Description": " Describes annotation in en ",
    "Descriptions": {
      "deferred": {
        "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Annotations('core.automobiles:Pressure108')/Descriptions"
      }
    }
  }
}
```

Request without the query parameter `$expand(/ThingConfiguration/v1/Annotations('core.automobiles:Pressure108'))?$expand=Descriptions` returns the following response:

```
{
  "d": {
    "_metadata": {
      "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Annotations('core.automobiles:Pressure108')",
      "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Annotations('core.automobiles:Pressure108')",
      "type": "com.sap.apptot.Annotation"
    },
    "Name": "core.automobiles:Pressure108",
    "PackageName": "core.automobiles",
    "Description": " Describes annotation in en ",
    "Descriptions": {
      "results": [
        {
          "_metadata": {
            "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Descriptions('en')",
            "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Descriptions('en')",
            "type": "com.sap.apptot.Description"
          },
          "LanguageCode": "en",
          "Description": " Describes annotation in en "
        },
        {
          "_metadata": {
            "id": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Descriptions('de')",
            "uri": "https://config-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/Descriptions('de')",
            "type": "com.sap.apptot.Description"
          },
          "LanguageCode": "de",

```

```
    "Description": "Describes annotation in de"  
  }  
]  
}  
}
```

### 8.2.7.3 Read all Annotations of a Package

With this method, you can retrieve all annotations of the specified package.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

#### Request

**URI:** `/ThingConfiguration/v1/Package('<package name>')/Annotations`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thingconf>.r`

#### Request Parameters

Parameter	Required	Data Type	Description
PackageName	Yes	String	Name of the package

#### Query Request Parameters

Parameter	Required	Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set.
<code>\$skip</code>	No	Integer	Number of the first n records to be excluded from the result set.

Parameter	Required	Type	Description
\$inlinecount	No	String	<p>Total number of annotations available; permissible values are <i>allpages</i> and <i>none</i>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"> <li>• If you use this with the \$top and \$skip parameters, the result contains the total count of annotations available.</li> <li>• If you use this with \$filter parameter, the result contains the total count of annotations based on the defined filter condition.</li> </ul>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; valid fields are name and description.</p> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; valid fields are name and description. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the separators – <i>and</i>, <i>or</i>.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using \$filter. Only valid field is annotation name.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are name, description, and package name.</p>

## Request Example

```
/ThingConfiguration/v1/Packages('core.automobiles')/Annotations
```

## Response

### Response Status and Error Codes

Code	Description
200	Annotations of the specified package retrieved successfully



## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Annotations('core.automobiles%3APressure108')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Annotations('core.automobiles%3APressure108')",
        "type": "com.sap.apptot.Annotation"
      },
      "Name": "core.automobiles:Pressure108",
      "PackageName": "core.automobiles",
      "Description": " Describes annotation in en ",
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Annotations('core.automobiles%3APressure108')/Descriptions"
        }
      }
    },
    {
      "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Annotations('core.automobiles%3APressure109')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Annotations('core.automobiles%3APressure109')",
        "type": "com.sap.apptot.Annotation"
      },
      "Name": "core.automobiles:Pressure109",
      "PackageName": "core.automobiles",
      "Description": " Describes annotation in en ",
      "Descriptions": {
        "__deferred": {
          "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Annotations('core.automobiles%3APressure109')/Descriptions"
        }
      }
    },
    {
      "__metadata": {
        "id": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Annotations('com.sap.apptot.analytics%3Adimension')",
        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Annotations('com.sap.apptot.analytics%3Adimension')",
        "type": "com.sap.apptot.Annotation"
      },
      "Name": "com.sap.apptot.analytics:dimension",
      "PackageName": "com.sap.apptot.analytics",
      "Description": null,
      "Descriptions": {
        "__deferred": {
```

```

        "uri": "https://config-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v1/
Annotations('com.sap.apptot.analytics%3Adimension')/Descriptions"
    }
}
]
}
}

```

## 8.2.7.4 Update an Annotation

With this method, you can update a specific annotation.

The **ETag** value is mandatory to update annotation. You must retrieve the **ETag** value of the annotation using the **GET (/ThingConfiguration/v1/Annotations('<annotation name>'))** service. The **ETag** value is assigned to the request header parameter **If-Match** in the **PATCH** request. If you do not include the **ETag** value in the request header or if the **ETag** value is not correct, the server displays an error message.

After updating annotation, the system generates a new **ETag** value for the annotation and replaces the old value with the new value.

### Note

You cannot change the name of the annotation using **PATCH** request.

## Request

**URI:** `/ThingConfiguration/v1/Annotations('<annotation name>')`

**Operation Type:** CRUD

**HTTP Method:** *PATCH*

**Permissions:** `<thingconf>.u`

### Request Headers

Header	Required	Values
If-Match	Yes	Latest ETag value of a specific annotation that is updated using the PATCH request.

### Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	String	Name of the annotation

### Request Example

`/ThingConfiguration/v1/Annotations('core.automobiles:Pressure108')`

## Payload

```
{
  "Name": "core.automobiles:Pressure108",
  "Descriptions": [
    {
      "LanguageCode": "en",
      "Description": "Describes annotation in en"
    },
    {
      "LanguageCode": "de",
      "Description": "Describes annotation in de"
    }
  ]
}
```

### Note

You cannot update the annotation name using the `PATCH` request. The annotation name specified in the request payload is ignored for update.

## Response

### Response Status and Error Codes

Code	Description
204	Annotations updated successfully

## 8.2.7.5 Delete an Annotation

With this method, you can delete the specified annotation.

The `ETag` value is mandatory to delete annotation. You must retrieve the `ETag` value of the annotation using the `GET (/ThingConfiguration/v1/Annotations('<annotation name>'))` service. The `ETag` value is assigned to the request header parameter `If-Match` in the [PATCH](#) request. If you do not include the `ETag` value in the request header or if the `ETag` value is not correct, the server displays an error message.

## Request

**URI:** `/ThingConfiguration/v1/Annotations('<annotation name>')`

**Operation Type:** CRUD

**HTTP Method:** [DELETE](#)

**Permissions:** `<thingconf>.d`

## Request Headers

Header	Required	Values
If-Match	Yes	Latest ETag value of a annotation that is updated using the <a href="#">PATCH</a> request.

## Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	String	Unique name of the annotation

## Request Example

```
/ThingConfiguration/v1/Annotations('core.automobiles:Pressure108')
```

## Response

### Response Status and Error Codes

Code	Description
204	Annotation deleted successfully

## 8.3 Event Configuration

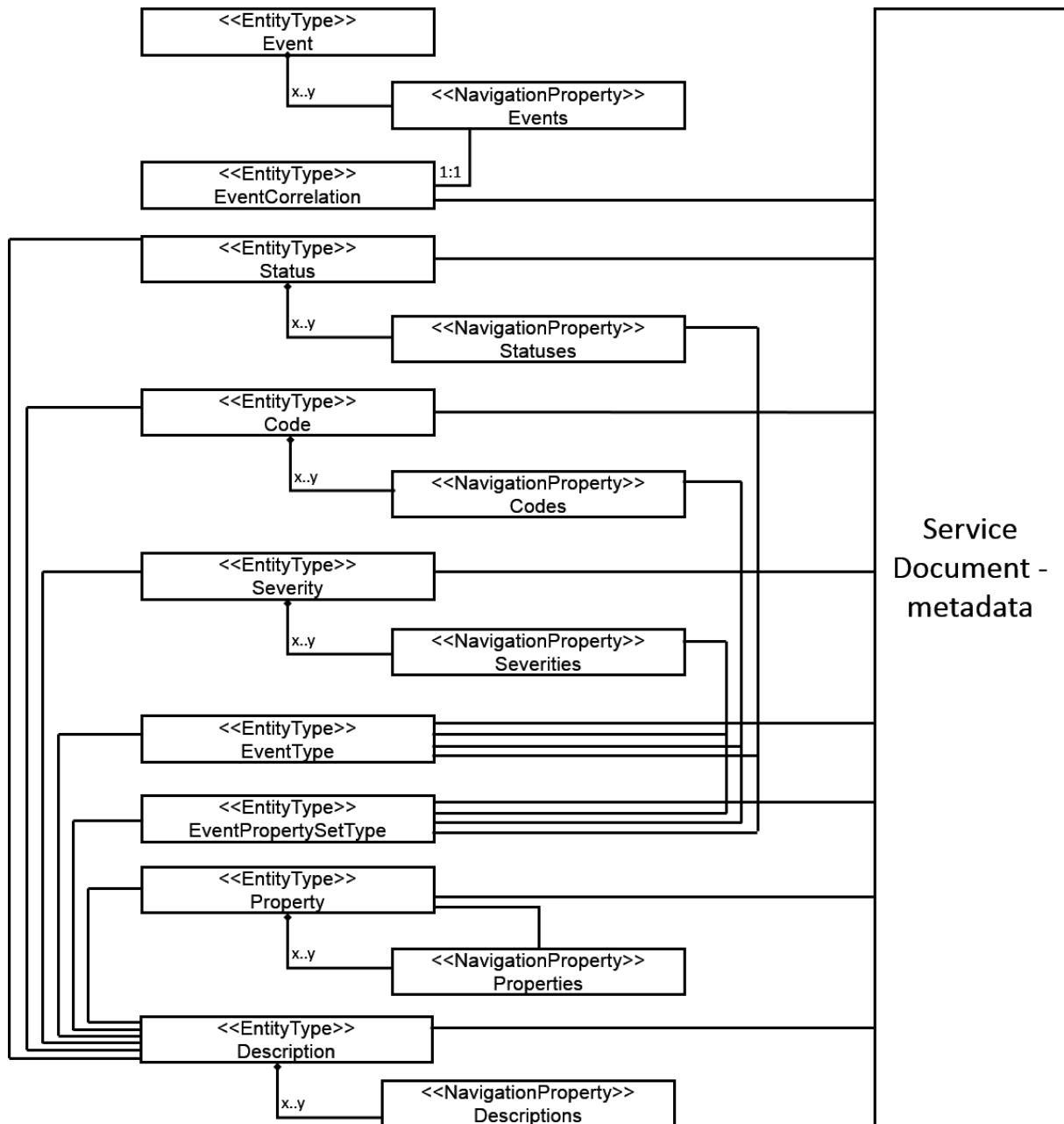
### Overview

With the event configuration service, you can create user-defined event types. The service provides a framework to design the user-defined event types containing all required information for IoT scenario. The event configuration service uses the package to define the metadata objects such as event type, event property set type, status, severity, and code. These objects are used to store event data. You can associate properties to event property set type and also assign a property set type to an event type.

**OData Version: 2.0**

### Entity Data Model

The following diagram shows the EDM of the Event Configuration service



Service Metadata URI: `http://<server address>[:<port number>]/ES/v1/$metadata`

## Resources

Resource	Description	Path
<a href="#">Event Type [page 520]</a>	Event Type	/v1/EventTypes

Resource	Description	Path
<a href="#">Event Property Set Type [page 557]</a>	Event Property Set Type	/v1/EventPropertySetTypes
<a href="#">User-defined Events based on OData Event Type [page 998]</a>	Events	/ES/EventType/<event type name>/v1/Events

## 8.3.1 Read Metadata of Event Configuration

With this method, you can retrieve the metadata of event configuration.

### Request

**URI:** /ES/v1/\$metadata

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <tde>.r

### Request Parameters

None

### Request Example

/ES/v1/\$metadata

### Response

#### Response Status and Error Codes

Code	Description
200	Event configuration metadata retrieved successfully.

### Payload

```
<?xml version="1.0" ?>
<edmx:Edmx Version="1.0" xmlns:edmx="http://schemas.microsoft.com/ado/2007/06/edmx">
```

```

    <edmx:DataServices m:DataServiceVersion="1.0" xmlns:m="http://
schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:sap="http://
www.sap.com/Protocols/SAPData">
    <Schema Namespace="com.sap.apptot.es" xmlns="http://
schemas.microsoft.com/ado/2008/09/edm">
    <EntityType Name="Status">
    <Key>
    <PropertyRef Name="EventStatus"></PropertyRef>
    </Key>
    <Property Name="EventStatus" Type="Edm.String"></Property>
    <Property Name="LanguageCode" Type="Edm.String"></Property>
    <Property Name="Description" Type="Edm.String"></Property>
    <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.es.Status_Description" FromRole="Statuses"
ToRole="Descriptions"></NavigationProperty>
    </EntityType>
    <EntityType Name="Annotations">
    <Key>
    <PropertyRef Name="Name"></PropertyRef>
    </Key>
    <Property Name="Name" Type="Edm.String" Nullable="false"
MaxLength="81" sap:creatable="true" sap:updatable="false">
    <Documentation>
    <LongDescription>Name of Annotation</LongDescription>
    </Documentation>
    </Property>
    <Property Name="PackageName" Type="Edm.String" Nullable="true"
MaxLength="50" sap:creatable="false" sap:updatable="false">
    <Documentation>
    <LongDescription>Package Name</LongDescription>
    </Documentation>
    </Property>
    <NavigationProperty Name="Annotations"
Relationship="com.sap.apptot.es.EventPropertySetType_Annotation"
FromRole="EventPropertySetTypes" ToRole="Annotations"></NavigationProperty>
    </EntityType>
    <EntityType Name="Properties">
    <Key>
    <PropertyRef Name="Name"></PropertyRef>
    <PropertyRef Name="PropertySetType"></PropertyRef>
    </Key>
    <Property Name="Name" Type="Edm.String" Nullable="false"></
Property>
    <Property Name="PropertySetType" Type="Edm.String"
Nullable="false"></Property>
    <Property Name="Type" Type="Edm.String"></Property>
    <Property Name="PropertyLength" Type="Edm.String"></Property>
    <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.es.Property_Description" FromRole="Properties"
ToRole="Descriptions"></NavigationProperty>
    </EntityType>
    <EntityType Name="EventTypes">
    <Key>
    <PropertyRef Name="Name"></PropertyRef>
    </Key>
    <Property Name="Name" Type="Edm.String" Nullable="false"></
Property>
    <Property Name="PackageName" Type="Edm.String"
Nullable="false"></Property>
    <Property Name="PropertySetId" Type="Edm.String"></Property>
    <Property Name="PropertySetType" Type="Edm.String"></Property>
    <Property Name="LanguageCode" Type="Edm.String"></Property>
    <Property Name="Description" Type="Edm.String"></Property>
    <Property Name="EventTypeState" Type="Edm.String"></Property>
    <NavigationProperty Name="PropertySetDescriptions"
Relationship="com.sap.apptot.es.EventType_PropertySetDescription"
FromRole="EventTypes" ToRole="PropertySetDescriptions"></NavigationProperty>

```

```

        <NavigationProperty Name="Statuses"
Relationship="com.sap.apptot.es.EventType_Status" FromRole="EventTypes"
ToRole="Statuses"></NavigationProperty>
        <NavigationProperty Name="Severities"
Relationship="com.sap.apptot.es.EventType_Severity" FromRole="EventTypes"
ToRole="Severities"></NavigationProperty>
        <NavigationProperty Name="Codes"
Relationship="com.sap.apptot.es.EventType_Code" FromRole="EventTypes"
ToRole="Codes"></NavigationProperty>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.es.EventType_Description" FromRole="EventTypes"
ToRole="Descriptions"></NavigationProperty>
    </EntityType>
    <EntityType Name="EventCorrelations">
        <Key>
            <PropertyRef Name="CorrelationId"></PropertyRef>
        </Key>
        <Property Name="CorrelationId" Type="Edm.String">
            <Documentation>
                <LongDescription>Event Correlation ID</LongDescription>
            </Documentation>
        </Property>
        <Property Name="EventId" Type="Edm.String"></Property>
        <Property Name="BusinessTimestamp" Type="Edm.String"></Property>
        <NavigationProperty Name="Events"
Relationship="com.sap.apptot.es.EventCorrelation_Event"
FromRole="EventCorrelations" ToRole="Events"></NavigationProperty>
    </EntityType>
    <EntityType Name="PropertySetDescriptions">
        <Key>
            <PropertyRef Name="LanguageCode"></PropertyRef>
        </Key>
        <Property Name="LanguageCode" Type="Edm.String" Nullable="false">
            <Documentation>
                <LongDescription>Language Code</LongDescription>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String">
            <Documentation>
                <LongDescription>Description</LongDescription>
            </Documentation>
        </Property>
    </EntityType>
    <EntityType Name="EventPropertySetTypes">
        <Key>
            <PropertyRef Name="Name"></PropertyRef>
        </Key>
        <Property Name="Name" Type="Edm.String" Nullable="false"></
Property>
        <Property Name="PackageName" Type="Edm.String"
Nullable="false"></Property>
        <Property Name="DataCategory" Type="Edm.String"></Property>
        <Property Name="Language" Type="Edm.String"></Property>
        <Property Name="PropertySetTypeDescription" Type="Edm.String"></
Property>
        <NavigationProperty Name="Properties"
Relationship="com.sap.apptot.es.EventPropertySetType_Property"
FromRole="EventPropertySetTypes" ToRole="Properties"></NavigationProperty>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.es.EventPropertySetType_Description"
FromRole="EventPropertySetTypes" ToRole="Descriptions"></NavigationProperty>
        <NavigationProperty Name="Annotations"
Relationship="com.sap.apptot.es.EventPropertySetType_Annotation"
FromRole="EventPropertySetTypes" ToRole="Annotations"></NavigationProperty>
    </EntityType>
    <EntityType Name="Descriptions">
        <Key>
            <PropertyRef Name="LanguageCode"></PropertyRef>

```



```

        </Key>
        <Property Name="LanguageCode" Type="Edm.String">
            <Documentation>
                <LongDescription>Language Code</LongDescription>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String">
            <Documentation>
                <LongDescription>Description</LongDescription>
            </Documentation>
        </Property>
    </EntityType>
    <EntityType Name="Severity">
        <Key>
            <PropertyRef Name="EventSeverity"></PropertyRef>
        </Key>
        <Property Name="EventSeverity" Type="Edm.String"></Property>
        <Property Name="LanguageCode" Type="Edm.String"></Property>
        <Property Name="Description" Type="Edm.String"></Property>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.appiot.es.Severity_Description" FromRole="Severities"
ToRole="Descriptions"></NavigationProperty>
    </EntityType>
    <EntityType Name="Events">
        <Key>
            <PropertyRef Name="EventId"></PropertyRef>
        </Key>
        <Property Name="EventId" Type="Edm.String" Nullable="false">
            <Documentation>
                <LongDescription>Event unique ID</LongDescription>
            </Documentation>
        </Property>
        <Property Name="CorrelationId" Type="Edm.String">
            <Documentation>
                <LongDescription>Event Correlation ID</LongDescription>
            </Documentation>
        </Property>
        <Property Name="BusinessTimestamp" Type="Edm.String"></Property>
        <Property Name="Type" Type="Edm.String"></Property>
        <Property Name="EventType" Type="Edm.String"></Property>
        <Property Name="EventStatus" Type="Edm.String"></Property>
        <Property Name="EventSeverity" Type="Edm.Int64"></Property>
        <Property Name="EventCode" Type="Edm.String"></Property>
        <Property Name="EventSource" Type="Edm.String"></Property>
        <Property Name="ThingId" Type="Edm.String"></Property>
        <Property Name="ThingProperty" Type="Edm.String"></Property>
        <Property Name="ExternalId" Type="Edm.String"></Property>
        <Property Name="EventInfo" Type="Edm.String"></Property>
    </EntityType>
    <EntityType Name="Codes">
        <Key>
            <PropertyRef Name="EventCode"></PropertyRef>
        </Key>
        <Property Name="EventCode" Type="Edm.String"></Property>
        <Property Name="LanguageCode" Type="Edm.String"></Property>
        <Property Name="Description" Type="Edm.String"></Property>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.appiot.es.Code_Description" FromRole="Codes"
ToRole="Descriptions"></NavigationProperty>
    </EntityType>
    <Association Name="EventPropertySetType_Annotation">
        <End Type="com.sap.appiot.es.EventPropertySetType"
Multiplicity="1" Role="EventPropertySetTypes"></End>
        <End Type="com.sap.appiot.es.Annotations" Multiplicity="*"
Role="Annotations"></End>
    </Association>
    <Association Name="Property_Description">

```

```

        <End Type="com.sap.apptot.es.Property" Multiplicity="1"
Role="Properties"></End>
        <End Type="com.sap.apptot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>
    <Association Name="Severity_Description">
        <End Type="com.sap.apptot.es.Severity" Multiplicity="1"
Role="Severities"></End>
        <End Type="com.sap.apptot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>
    <Association Name="EventType_Code">
        <End Type="com.sap.apptot.es.EventType" Multiplicity="1"
Role="EventTypes"></End>
        <End Type="com.sap.apptot.es.Code" Multiplicity="*"
Role="Codes"></End>
    </Association>
    <Association Name="EventPropertySetType_Property">
        <End Type="com.sap.apptot.es.EventPropertySetType"
Multiplicity="1" Role="EventPropertySetTypes"></End>
        <End Type="com.sap.apptot.es.Property" Multiplicity="*"
Role="Properties"></End>
    </Association>
    <Association Name="EventType_Status">
        <End Type="com.sap.apptot.es.EventType" Multiplicity="1"
Role="EventTypes"></End>
        <End Type="com.sap.apptot.es.Status" Multiplicity="*"
Role="Statuses"></End>
    </Association>
    <Association Name="EventCorrelation_Event">
        <End Type="com.sap.apptot.es.EventCorrelation" Multiplicity="1"
Role="EventCorrelations"></End>
        <End Type="com.sap.apptot.es.Event" Multiplicity="*"
Role="Events"></End>
    </Association>
    <Association Name="EventType_Description">
        <End Type="com.sap.apptot.es.EventType" Multiplicity="1"
Role="EventTypes"></End>
        <End Type="com.sap.apptot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>
    <Association Name="EventType_Severity">
        <End Type="com.sap.apptot.es.EventType" Multiplicity="1"
Role="EventTypes"></End>
        <End Type="com.sap.apptot.es.Severity" Multiplicity="*"
Role="Severities"></End>
    </Association>
    <Association Name="Status_Description">
        <End Type="com.sap.apptot.es.Status" Multiplicity="1"
Role="Statuses"></End>
        <End Type="com.sap.apptot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>
    <Association Name="Code_Description">
        <End Type="com.sap.apptot.es.Code" Multiplicity="1"
Role="Codes"></End>
        <End Type="com.sap.apptot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>
    <Association Name="EventPropertySetType_Description">
        <End Type="com.sap.apptot.es.EventPropertySetType"
Multiplicity="1" Role="EventPropertySetTypes"></End>
        <End Type="com.sap.apptot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>
    <Association Name="EventType_PropertySetDescription">
        <End Type="com.sap.apptot.es.EventType" Multiplicity="1"
Role="EventTypes"></End>

```

```

        <End Type="com.sap.appiot.es.PropertySetDescription"
Multiplicity="*" Role="PropertySetDescriptions"></End>
    </Association>
    <EntityContainer Name="UnifiedEvents"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="Statuses"
EntityType="com.sap.appiot.es.Status"></EntitySet>
        <EntitySet Name="Annotations"
EntityType="com.sap.appiot.es.Annotations"></EntitySet>
        <EntitySet Name="Properties"
EntityType="com.sap.appiot.es.Property"></EntitySet>
        <EntitySet Name="EventTypes"
EntityType="com.sap.appiot.es.EventType"></EntitySet>
        <EntitySet Name="EventCorrelations"
EntityType="com.sap.appiot.es.EventCorrelation"></EntitySet>
        <EntitySet Name="PropertySetDescriptions"
EntityType="com.sap.appiot.es.PropertySetDescription"></EntitySet>
        <EntitySet Name="EventPropertySetTypes"
EntityType="com.sap.appiot.es.EventPropertySetType"></EntitySet>
        <EntitySet Name="Descriptions"
EntityType="com.sap.appiot.es.Description"></EntitySet>
        <EntitySet Name="Severities"
EntityType="com.sap.appiot.es.Severity"></EntitySet>
        <EntitySet Name="Events" EntityType="com.sap.appiot.es.Event"></
EntitySet>
        <EntitySet Name="Codes" EntityType="com.sap.appiot.es.Code"></
EntitySet>
        <AssociationSet Name="EventPropertySetType_Annotation"
Association="com.sap.appiot.es.EventPropertySetType_Annotation">
            <End EntitySet="EventPropertySetTypes"
Role="EventPropertySetTypes"></End>
            <End EntitySet="Annotations" Role="Annotations"></End>
        </AssociationSet>
        <AssociationSet Name="Property_Description"
Association="com.sap.appiot.es.Property_Description">
            <End EntitySet="Properties" Role="Properties"></End>
            <End EntitySet="Descriptions" Role="Descriptions"></End>
        </AssociationSet>
        <AssociationSet Name="Severity_Description"
Association="com.sap.appiot.es.Severity_Description">
            <End EntitySet="Severities" Role="Severities"></End>
            <End EntitySet="Descriptions" Role="Descriptions"></End>
        </AssociationSet>
        <AssociationSet Name="EventType_Code"
Association="com.sap.appiot.es.EventType_Code">
            <End EntitySet="EventTypes" Role="EventTypes"></End>
            <End EntitySet="Codes" Role="Codes"></End>
        </AssociationSet>
        <AssociationSet Name="EventPropertySetType_Property"
Association="com.sap.appiot.es.EventPropertySetType_Property">
            <End EntitySet="EventPropertySetTypes"
Role="EventPropertySetTypes"></End>
            <End EntitySet="Properties" Role="Properties"></End>
        </AssociationSet>
        <AssociationSet Name="EventType_Status"
Association="com.sap.appiot.es.EventType_Status">
            <End EntitySet="EventTypes" Role="EventTypes"></End>
            <End EntitySet="Statuses" Role="Statuses"></End>
        </AssociationSet>
        <AssociationSet Name="EventCorrelation_Event"
Association="com.sap.appiot.es.EventCorrelation_Event">
            <End EntitySet="EventCorrelations"
Role="EventCorrelations"></End>
            <End EntitySet="Events" Role="Events"></End>
        </AssociationSet>
        <AssociationSet Name="EventType_Description"
Association="com.sap.appiot.es.EventType_Description">
            <End EntitySet="EventTypes" Role="EventTypes"></End>

```

```

        <End EntitySet="Descriptions" Role="Descriptions"></End>
    </AssociationSet>
    <AssociationSet Name="EventType_Severity"
Association="com.sap.apptot.es.EventType_Severity">
        <End EntitySet="EventTypes" Role="EventTypes"></End>
        <End EntitySet="Severities" Role="Severities"></End>
    </AssociationSet>
    <AssociationSet Name="Status_Description"
Association="com.sap.apptot.es.Status_Description">
        <End EntitySet="Statuses" Role="Statuses"></End>
        <End EntitySet="Descriptions" Role="Descriptions"></End>
    </AssociationSet>
    <AssociationSet Name="Code_Description"
Association="com.sap.apptot.es.Code_Description">
        <End EntitySet="Codes" Role="Codes"></End>
        <End EntitySet="Descriptions" Role="Descriptions"></End>
    </AssociationSet>
    <AssociationSet Name="EventPropertySetType_Description"
Association="com.sap.apptot.es.EventPropertySetType_Description">
        <End EntitySet="EventPropertySetTypes"
Role="EventPropertySetTypes"></End>
        <End EntitySet="Descriptions" Role="Descriptions"></End>
    </AssociationSet>
    <AssociationSet Name="EventType_PropertySetDescription"
Association="com.sap.apptot.es.EventType_PropertySetDescription">
        <End EntitySet="EventTypes" Role="EventTypes"></End>
        <End EntitySet="PropertySetDescriptions"
Role="PropertySetDescriptions"></End>
    </AssociationSet>
</EntityContainer>
</Schema>
</edmx:DataServices>
</edmx:Edmx>

```

## Related Information

[Event Configuration \[page 512\]](#)

### 8.3.2 Event Type

You can use the event configuration service to perform the following:

- Configure user-defined event types along with their attributes
- Create events based on the predefined and user-defined event types
- Retrieve event details for a specific event type

You can create event types with one of the following states:

- **Mutable:** You can delete events that are created based on an event type with `EventTypeState=Mutable`.
- **Immutable:** You cannot delete events that are created based on an event type with `EventTypeState=Immutable`.

**Resource Path:** `http://<server address>[:<port number>][<path>]/ES/v1`

**Example for a base URI in a cloud foundry environment:** `https://events-sap.cfapps.eu10.hana.ondemand.com/ES`

## Note

`com.sap.appiot.eventtypes:StandardEventType` is a predefined event type delivered by SAP. The values for the property such as `EventStatus` and `EventSeverity` are already defined and you cannot assign different values for these properties.

Only the following values are supported for `EventStatus` of a predefined event type:

- *Open*
- *InProcess*
- *Completed*

Only values *1*, *2*, and *3* are supported for `EventSeverity` of a predefined event type.

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scope
<i>POST</i>	<a href="#">Create Event Type [page 522]</a>	/ES/v1/EventTypes	<tde>.c
<i>GET</i>	<a href="#">Read Details of an Event Type [page 527]</a>	/ES/v1/EventTypes ('<event type name>')	<tde>.r
<i>GET</i>	<a href="#">Read all Event Types [page 542]</a>	/ES/v1/EventTypes	<tde>.r
<i>PATCH</i>	<a href="#">Update an Event Type [page 538]</a>	/ES/v1/EventTypes ('<event type name>')	<tde>.u
<i>DELETE</i>	<a href="#">Delete an Event Type [page 541]</a>	/ES/v1/EventTypes ('<event type name>')	<tde>.d

## Event Type Object Properties

Property	Type	Mandatory	Maximum Length	Description
Name	String	Yes		Name of the event type
Description	String	No	255	Description of the event type
PackageName	String	Yes	50	Name of the package for which the event types are configured
PropertySetID		Yes	30	Name of the property set
PropertySetType		Yes	81	Name of the property set type

Property	Type	Mandatory	Maximum Length	Description
PropertySetDescriptions	String	No	60	Description of the property set for each language with the corresponding ISO language code
EventTypeState	String	No	Fixed	State of the event. The possible values are <i>Mutable</i> and <i>Immutable</i> . Default value is <i>Immutable</i> .
EventStatus	String	No	50	Status of the event
EventSeverity	Integer	No	-	Severity of the event
EventCode	String	No	50	Code of the event

### Event Type Description Properties

Property	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the object description
Description	String	255	Description of the object

## Related Information

[Event Property Set Type \[page 557\]](#)

[User-defined Events based on OData Event Type \[page 998\]](#)

### 8.3.2.1 Create Event Type

With this method, you can configure user-defined event types.

The event type configuration definition must contain at least one event type and event property set type. You must associate only one event property set type as property set for an event type. An event type cannot have multiple property sets. The fields *EventStatus*, *EventSeverity*, and *EventCode* are optional. However, if you define enumeration values for these fields while creating an event type, you must assign only one of these defined enumeration values for these fields while creating an event. If you assign different values, the system displays an error message.

#### Note

To indicate that the fields *EventStatus*, *EventSeverity*, and *EventCode* are optional, you must include them with an empty array in the request payload.

## Naming convention for event type

- Always prefixed with the package name.
- Can be a combination of alphanumeric characters and underscore.
- Each name must be fully qualified and contains a package name, colon as a separator and the object name. Maximum length of fully qualified name is 81 characters.
- Do not start the object name with a digit or an underscore.
- Length should be at least 3 characters.
- Camel case is generally allowed, however there is a uniqueness check carried out against textual case conflicts:
  - Two names with same characters of different textual case is not allowed. For example: A value "AbcDef" is not accepted when there is another value "abcDEF" that exists.
- Verify that a colon is used only to separate a package name and the object name. The syntax is: <package>:<name>. For example: core.automobiles:FlyWheel
- Name must be unique within a package.

## Naming convention for property set

- Only the following characters are supported:
  - Uppercase and lowercase alphabets *a* through *z* and *A* through *Z*
  - Numeric digits *0* through *9*
  - Punctuation marks underscore (*\_*), hyphen (*-*), colon (*:*), and full-stop (*.*)
- Maximum length is 30 characters.
- Property set names have to be unique for a given set of thing types.

## Request

**URI:** /ES/v1/EventTypes

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** <td>.c

### Request Parameters

None

### Request Example

/ES/v1/EventTypes

## Payload

**Format:** *JSON*

### Example 1

The following request payload is used to create an event type. As the property `EventTypeState` is not specified in this payload, all events created using this event type are immutable.

```
{
  "Name": "core.automobiles:ExceededTemperature",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Event type for exceeded temperature"
  },
  {
    "LanguageCode": "de",
    "Description": "Ereignistyp für überschrittene Temperatur"
  }
],
  "PackageName": "core.automobiles",
  "PropertySetId": "EngineTemp",
  "PropertySetType": "core.automobiles:Engine",
  "PropertySetDescriptions": [{
    "LanguageCode": "en",
    "Description": "Engine property set type"
  }],
  "Statuses": [{
    "EventStatus": "Open",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Open Status"
    }]
  },
  {
    "EventStatus": "InProgress",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "In process Status"
    }]
  },
  {
    "EventStatus": "Completed",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Completed Status"
    }]
  }
],
  "Severities": [{
    "EventSeverity": 1,
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "High"
    }]
  },
  {
    "EventSeverity": 2,
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Medium"
    }]
  },
  {
    "EventSeverity": 3,
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Low"
    }]
  }
],
  "Codes": [{
    "EventCode": "EQ12",
```



```

        "Descriptions": [{
            "LanguageCode": "en",
            "Description": "Event Code 12"
        }]
    }
}

```

## Example 2

The following request payload is to create an event type. As the property `EventTypeState="Mutable"` is specified in this payload, all events created using this event type are mutable.

```

{
  "Name": "core.automobiles:ExceededPressure",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Event type for exceeded pressure"
  }],
  "PackageName": "core.automobiles",
  "EventTypeState": "Mutable",
  "PropertySetId": "WheelPressure",
  "PropertySetType": "core.automobiles:Wheel",
  "PropertySetDescriptions": [{
    "LanguageCode": "en",
    "Description": "Wheel property set type"
  }],
  "Statuses": [{
    "EventStatus": "Open",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Open Status"
    }]
  },
  {
    "EventStatus": "InProgress",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "In process Status"
    }]
  },
  {
    "EventStatus": "Completed",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Completed Status"
    }]
  }
],
  "Severities": [{
    "EventSeverity": 1,
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "High"
    }]
  },
  {
    "EventSeverity": 2,
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Medium"
    }]
  },
  {
    "EventSeverity": 3,
    "Descriptions": [{
      "LanguageCode": "en",

```

```

        "Description": "Low"
    }
  ]
},
{
  "Codes": [{
    "EventCode": "EQ12",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Event Code 12"
    }]
  }]
}
}

```

## Response

### Response Status and Error Codes

Code	Description
201	Event type created successfully

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as illustrated in the following examples.

The following is the response payload after creating an event type `core.automobiles:ExceededTemperature`.

```

{
  "d": {
    "__metadata": {
      "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
      "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
      "type": "com.sap.apptot.es.EventTypes"
    },
    "Name": "core.automobiles:ExceededTemperature",
    "PackageName": "core.bbautomobiles",
    "EventTypeState": "",
    "PropertySetId": "EngineTemp",
    "PropertySetType": "core.automobiles:Engine",
    "LanguageCode": "",
    "Description": "",
    "PropertySetDescriptions": {
      "__deferred": {
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')/PropertySetDescriptions"
      }
    },
    "Statuses": {
      "__deferred": {

```

```

        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')/Statuses"
    },
    "Severities": {
        "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')/Severities"
        }
    },
    "Codes": {
        "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')/Codes"
        }
    },
    "Descriptions": {
        "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')/Descriptions"
        }
    }
}
}
}

```

## Related Information

[Event Type \[page 520\]](#)

### 8.3.2.2 Read Details of an Event Type

With this method, you can retrieve the details of the specified event type.

#### Request

**URI:** `/ES/v1/EventTypes('<event type name>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

#### Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	String	Name of the event type

#### Request Example

```
/ES/v1/EventTypes('core.automobiles:ExceededTemperature')
```

Retrieves details of the event type `core.automobiles:ExceededTemperature`

## Response

### Response Status and Error Codes

Code	Description
200	Details of the event type are retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as depicted in the following examples.

### Example 1

`/ES/v1/EventTypes('core.automobiles:ExceededTemperature')`

```
{
  "d": {
    "__metadata": {
      "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
      "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
      "type": "com.sap.apptot.es.EventTypes"
    },
    "Name": "core.automobiles:ExceededTemperature",
    "PackageName": "core.automobiles",
    "EventTypeState": "",
    "PropertySetId": "EngineTemp",
    "PropertySetType": "core.automobiles:Temperature",
    "LanguageCode": "",
    "Description": "",
    "PropertySetDescriptions": {
      "__deferred": {
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')/PropertySetDescriptions"
      }
    },
    "Statuses": {
      "__deferred": {
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')/Statuses"
      }
    },
    "Severities": {
      "__deferred": {
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')/Severities"
      }
    },
    "Codes": {
```

```

        "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')/Codes"
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')/Descriptions"
            }
        }
    }
}

```

## Example 2

/ES/v1/EventTypes('core.automobiles:ExceededTemperature')/Descriptions

Retrieves descriptions of the specified event type

```

{
    "d": {
        "results": [
            {
                "__metadata": {
                    "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/Descriptions('en')",
                    "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/Descriptions('en')",
                    "type": "com.sap.apptot.es.Descriptions"
                },
                "LanguageCode": "en",
                "Description": "Event type for exceeded temperature"
            },
            {
                "__metadata": {
                    "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/Descriptions('de')",
                    "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/Descriptions('de')",
                    "type": "com.sap.apptot.es.Descriptions"
                },
                "LanguageCode": "de",
                "Description": "Ereignistyp für überschrittene Temperatur"
            }
        ]
    }
}

```

## Related Information

[Event Type \[page 520\]](#)

## 8.3.2.3 Read Statuses of an Event Type

With this method, you can retrieve statuses of the specified event type.

### Request

**URI:** `/ES/v1/EventTypes('<event type name>')/Statuses`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

### Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	String	Name of the event type

### Request Example

```
/ES/v1/EventTypes('core.automobiles:ExceededTemperature')/Statuses
```

### Response

#### Response Status and Error Codes

Code	Description
200	Statuses of the specified event type retrieved successfully.

### Payload

**Format:** [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as depicted in the following examples.

#### Example 1

```
/ES/v1/EventTypes('core.automobiles:ExceededTemperature')/Statuses
```

Retrieves statuses of the specified event type

```
{
  "d": {
    "results": [
      {
```

```

        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('Open')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('Open')",
          "type": "com.sap.apptot.es.Status"
        },
        "EventStatus": "Open",
        "LanguageCode": "en",
        "Description": "Open Status",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('Open')/Descriptions"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('InProgress')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('InProgress')",
          "type": "com.sap.apptot.es.Status"
        },
        "EventStatus": "InProgress",
        "LanguageCode": "en",
        "Description": "In process Status",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('InProgress')/Descriptions"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('Completed')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('Completed')",
          "type": "com.sap.apptot.es.Status"
        },
        "EventStatus": "Completed",
        "LanguageCode": "en",
        "Description": "Completed Status",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('Completed')/Descriptions"
          }
        }
      }
    ]
  }
}

```

## Example 2

```

/ES/v1/EventTypes('core.automobiles:ExceededTemperature')/Statuses?
$expand=Descriptions

```

Retrieves statuses along with the expanded list of descriptions for the specified event type

```

{
  "d": {
    "results": [

```

```

    {
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('Open')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('Open')",
        "type": "com.sap.apptot.es.Status"
      },
      "EventStatus": "Open",
      "LanguageCode": null,
      "Description": null,
      "Descriptions": {
        "results": [
          {
            "__metadata": {
              "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('en')",
              "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('en')",
              "type": "com.sap.apptot.es.Descriptions"
            },
            "LanguageCode": "en",
            "Description": "Open Status"
          }
        ]
      }
    },
    {
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('InProgress')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('InProgress')",
        "type": "com.sap.apptot.es.Status"
      },
      "EventStatus": "InProgress",
      "LanguageCode": null,
      "Description": null,
      "Descriptions": {
        "results": [
          {
            "__metadata": {
              "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('en')",
              "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('en')",
              "type": "com.sap.apptot.es.Descriptions"
            },
            "LanguageCode": "en",
            "Description": "In process Status"
          }
        ]
      }
    },
    {
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('Completed')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Statuses('Completed')",
        "type": "com.sap.apptot.es.Status"
      },
      "EventStatus": "Completed",
      "LanguageCode": null,
      "Description": null,
      "Descriptions": {
        "results": [
          {

```



```

        "__metadata": {
          "id": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('en')",
          "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('en')",
          "type": "com.sap.apptot.es.Descriptions"
        },
        "LanguageCode": "en",
        "Description": "Completed Status"
      }
    ]
  }
}

```

## Related Information

[Event Type \[page 520\]](#)

### 8.3.2.4 Read Severities of an Event Type

With this method, you can retrieve severities of the specified event type.

## Request

**URI:** /ES/v1/EventTypes('<event type name>')/Severities

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <tde>.r

## Request Parameters

Parameter	Required	Data Type	Description
event type name	Yes	String	Name of the event type

## Request Example

```
/ES/v1/EventTypes('core.automobiles:ExceededTemperature')/Severities
```

## Response

### Response Status and Error Codes

Code	Description
200	Severities of the specified event type retrieved successfully.

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples.

### Example 1

/ES/v1/EventTypes('core.automobiles:ExceededTemperature')/Severities

Retrieves severities of the specified event type

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('1')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('1')",
          "type": "com.sap.apptot.es.Severity"
        },
        "EventSeverity": "1",
        "LanguageCode": "en",
        "Description": "High",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('1')/Descriptions"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('2')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('2')",
          "type": "com.sap.apptot.es.Severity"
        },
        "EventSeverity": "2",
        "LanguageCode": "en",
        "Description": "Medium",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('2')/Descriptions"
          }
        }
      }
    ]
  }
}
```

```

        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('3')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('3')",
          "type": "com.sap.apptot.es.Severity"
        },
        "EventSeverity": "3",
        "LanguageCode": "en",
        "Description": "Low",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('3')/Descriptions"
          }
        }
      }
    ]
  }
}

```

## Example 2

```

/ES/v1/EventTypes('core.automobiles:ExceededTemperature')/Severities?
$expand=Descriptions

```

Retrieves severities along with the expanded list of descriptions for the specified event type

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('1')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('1')",
          "type": "com.sap.apptot.es.Severity"
        },
        "EventSeverity": "1",
        "LanguageCode": "en",
        "Description": "High",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('1')/Descriptions"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('2')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('2')",
          "type": "com.sap.apptot.es.Severity"
        },
        "EventSeverity": "2",
        "LanguageCode": "en",
        "Description": "Medium",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('2')/Descriptions"
          }
        }
      }
    ]
  }
}

```

```

    {
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('3')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('3')",
        "type": "com.sap.apptot.es.Severity"
      },
      "EventSeverity": "3",
      "LanguageCode": "en",
      "Description": "Low",
      "Descriptions": {
        "__deferred": {
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Severities('3')/Descriptions"
        }
      }
    }
  ]
}

```

## Related Information

[Event Type \[page 520\]](#)

### 8.3.2.5 Read Codes of an Event Type

With this method, you can retrieve codes of the specified event type.

#### Request

**URI:** /ES/v1/EventTypes('<event type name>')/Codes

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <tde>.r

#### Request Parameters

Parameter	Required	Data Type	Description
event type name	Yes	String	Name of the event type

#### Request Example

/ES/v1/EventTypes('core.automobiles:ExceededTemperature')/Codes

## Response

### Response Status and Error Codes

Code	Description
200	Codes of the specified event type retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as illustrated in the following examples.

### Example 1

```
/ES/v1/EventTypes('core.automobiles:ExceededTemperature')/Codes
```

Retrieves codes of the specified event type

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Codes('EQ12')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Codes('EQ12')",
          "type": "com.sap.apptot.es.Codes"
        },
        "EventCode": "EQ12",
        "LanguageCode": "en",
        "Description": "Event Code 12",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Codes('EQ12')/Descriptions"
          }
        }
      }
    ]
  }
}
```

### Example 2

```
/ES/v1/EventTypes('core.automobiles:ExceededTemperature')/Codes?
```

```
$expand=Descriptions
```

Retrieves codes along with its expanded list of descriptions for the specified event type.

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Codes('EQ12')",

```

```

        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Codes('EQ12')",
        "type": "com.sap.apptot.es.Codes"
    },
    "EventCode": "EQ12",
    "LanguageCode": null,
    "Description": null,
    "Descriptions": {
        "results": [
            {
                "__metadata": {
                    "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('en')",
                    "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('en')",
                    "type": "com.sap.apptot.es.Descriptions"
                },
                "LanguageCode": "en",
                "Description": "Event Code 12"
            }
        ]
    }
}

```

## Related Information

[Event Type \[page 520\]](#)

### 8.3.2.6 Update an Event Type

With this method, you can update a specific event type.

The *ETag* value is mandatory to update an event type. You must retrieve the *ETag* value of the event type using the GET (/ES/v1/EventTypes('<event type name>')) service. The *ETag* value is assigned to the request header parameter If-Match in the PATCH request. If you do not include the *ETag* value in the request header or if the *ETag* value is not correct, the server displays an error message.

#### Note

- You can only update the already existing description of statuses, severities, and codes.
- You can add enumeration values for the fields *EventStatus*, *EventSeverity*, and *EventCode* only if the following conditions are met:
  - The event type was created with an empty array for the fields *EventStatus*, *EventSeverity*, and *EventCode*
  - There are no events created for the event type.
- You can indicate the fields *EventStatus*, *EventSeverity*, and *EventCode* as optional by using an empty array, irrespective of events created or not created for the event type.
- You can delete the already defined enumeration values for the fields *EventStatus*, *EventSeverity*, and *EventCode* of an event type, if there are no events created for that event type.

- You can add descriptions for an event type in different languages. If you want to remove the description in a specific language, assign null value for the **Description** field.
- If an event is already created for an event type, the fields [EventStatus](#), [EventSeverity](#), and [EventCode](#) can be updated as explained in the following table:

EventStatus	EventSeverity	EventCode	What can be updated?
No enumeration values defined in event type configuration			Cannot update these fields by adding new enumeration values
Enumeration values defined in event type configuration	No enumeration values defined in event type in event type configuration	No enumeration values defined in event type in event type configuration	You can add new enumeration values only for the field <a href="#">EventStatus</a>
No enumeration values defined in event type in event type configuration	Enumeration values defined in event type configuration	No enumeration values defined in event type in event type configuration	You can add new enumeration values only for the field <a href="#">EventSeverity</a>
No enumeration values defined in event type in event type configuration	No enumeration values defined in event type in event type configuration	Enumeration values defined in event type configuration	You can add new enumeration values only for the field <a href="#">EventCode</a>
Enumeration values defined in event type configuration	Enumeration values defined in event type configuration	Enumeration values defined in event type configuration	You can add new enumeration values only for the fields <a href="#">EventStatus</a> , <a href="#">EventSeverity</a> , <a href="#">EventCode</a>

- You can add, edit, or delete the following for an event type if there are no events already created for the event type.
  - EventTypeState
  - PropertySetID
  - PropertySetType
- The valid operations are `ADD`, `DELETE`, and `UPDATE`
- You cannot change the name of the event type using `PATCH` request.

After updating the event type, the system generates a new [ETag](#) value for the event type and replaces the old value with the new value.

## Request

**URI:** `/ES/v1/EventTypes('<event type name>')`

**Operation Type:** `CRUD`

**HTTP Method:** [PATCH](#)

**Permissions:** `<td>.u`

## Request Header Parameters

Header	Required	Values
If-Match	Yes	Latest <a href="#">ETag</a> value of a specific event type that is updated using the PATCH request.

## Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	String	Name of the event type

## Request Example

```
/ES/v1/EventTypes('core.automobiles:ExceededTemperature')
```

## Response

### Response Status and Error Codes

Code	Description
200	Event type updated successfully.

## Payload

Format: *JSON*

### Example 1

The following request payload updates the event type `core.automobiles:ExceededTemperature` by adding a new event code `EQ14`:

```
{
  "Name": "core.automobiles:ExceededTemperature",
  "PackageName": "core.automobiles",
  "Codes": [
    {
      "EventCode": "EQ14",
      "Operation": "ADD",
      "Descriptions": [{
        "LanguageCode": "en",
        "Description": "Event Code 14"
      }]
    }
  ]
}
```

### Example 2

The following request payload updates the event type `core.automobiles:ExceededTemperature` with the following:



- New event code EQ14 added.
- Empty arrays for *Statuses* and *Severities* indicate the fields *EventStatus* and *EventSeverity* accept free text or optional fields.

by adding a new event code and indicating the fields *EventStatuses* and *EventSeverities* :

```
{
  "Name": "core.automobiles:ExceededTemperature",
  "PackageName": "core.automobiles",
  "Statuses": [],
  "Severities": [],
  "Codes": [
    {
      "EventCode": "EQ14",
      "Operation": "ADD",
      "Descriptions": [
        {
          "LanguageCode": "en",
          "Description": "Event Code 14"
        }
      ]
    }
  ]
}
```

## Related Information

[Event Type \[page 520\]](#)

### 8.3.2.7 Delete an Event Type

With this method, you can delete an event type with a specific name.

The *ETag* value is mandatory to delete an event type. You must retrieve the *ETag* value of the event type using the GET (/ES/v1/EventTypes('<event type name>')) service. The *ETag* value is assigned to the request header parameter If-Match in the PATCH request. If you do not include the *ETag* value in the request header or if the *ETag* value is not correct, the server displays an error message.

#### i Note

- You cannot delete an event type if you have already created events for the event type.
- If you delete an event type, the system deletes the associated status, severity, and code of the event type.

## Request

**URI:** /ES/v1/EventTypes('<event type name>')

**Operation Type:** CRUD

HTTP Method: [DELETE](#)

Permissions: <td>.d

## Request Headers

Header	Required	Values
If-Match	Yes	Latest <a href="#">ETag</a> value of a specific event type that is updated using the PATCH request.

## Request Parameters

Parameter	Required	Data Type	Description
event_type_name	Yes	String	Name of the event type

## Request Example

```
/ES/v1/EventTypes('core.automobiles:ExceededTemperature')
```

## Response

### Response Status and Error Codes

Code	Description
204	Specified event type deleted successfully.

## Related Information

[Event Type \[page 520\]](#)

## 8.3.2.8 Read all Event Types

Retrieves a list of event types from the database

With this method, you send a request to the server to retrieve a subset of all event types, according to the filter criteria provided. You can restrict the set of event types to be retrieved by filter criteria based on the event type name, package name, property type name, and property set type name.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** /ES/v1/EventTypes

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <td>.r

## Request Parameters

None

## Query Request Parameters

Parameter	Type	Required	Description
\$top	Integer	No	Number of records to include in the result set.
\$skip	Integer	No	Number of records to exclude from the result set.
\$orderby	String	No	Field name to be used for sorting the result set in ascending order; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , and <a href="#">PropertySetType</a> .  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$select	String	No	Select condition for the data fields to be included in the result set; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , and <a href="#">PropertySetType</a> .
\$filter	String	No	Filter condition to be applied; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , and <a href="#">PropertySetType</a> .

## Request Example

/ES/v1/EventTypes

Retrieves all event types to which the user has access authorization

## Response

### Response Status and Error Codes

Code	Description
200	Retrieved all event types successfully.

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples.

### Example 1

/ES/v1/EventTypes

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')",
        "type": "com.sap.apptot.es.EventTypes"
      },
      "Name": "core.automobiles:ExceededPressure",
      "PackageName": "core.automobiles",
      "EventTypeState": "Mutable",
      "PropertySetId": "WheelPressure",
      "PropertySetType": "core.automobiles:Wheel",
      "LanguageCode": "en",
      "Description": "Event type for exceeded pressure",
      "PropertySetDescriptions": {
        "__deferred": {
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')/PropertySetDescriptions"
        }
      },
      "Statuses": {
        "__deferred": {
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')/Statuses"
        }
      },
      "Severities": {
        "__deferred": {
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')/Severities"
        }
      },
      "Codes": {
        "__deferred": {
```

```

        "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededPressure')/Codes"
    },
    "Descriptions": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededPressure')/Descriptions"
        }
    },
    {
        "__metadata": {
            "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
            "type": "com.sap.apptot.es.EventTypes"
        },
        "Name": "core.automobiles:ExceededTemperature",
        "PackageName": "core.automobiles",
        "EventTypeState": "",
        "PropertySetId": "EngineTemp",
        "PropertySetType": "core.automobiles:Engine",
        "LanguageCode": "en",
        "Description": "Event type for exceeded temperature",
        "PropertySetDescriptions": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/PropertySetDescriptions"
            }
        },
        "Statuses": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Statuses"
            }
        },
        "Severities": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Severities"
            }
        },
        "Codes": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Codes"
            }
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Descriptions"
            }
        }
    }
}
]
}
}

```

## Related Information

[Event Type \[page 520\]](#)

### 8.3.2.9 Read Statuses of all Event Types

With this method, you can retrieve statuses of all event types.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

#### Request

**URI:** `/ES/v1/EventTypes?$expand=Statuses`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

#### Request Parameters

None

#### Request Example

`/ES/v1/EventTypes?$expand=Statuses`

#### Query Request Parameters

Parameter	Type	Required	Description
<code>\$top</code>	Integer	No	Number of records to include in the result set.
<code>\$skip</code>	Integer	No	Number of records to exclude from the result set.

Parameter	Type	Required	Description
\$orderby	String	No	Field name to be used for sorting the result set in ascending order; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> and <a href="#">EventStatus</a> .  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$select	String	No	Select condition for the data fields to be included in the result set; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> and <a href="#">EventStatus</a> .
\$filter	String	No	Filter condition to be applied; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> and <a href="#">EventStatus</a> .

## Response

### Response Status and Error Codes

Code	Description
200	Statuses of all event types retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')",
```

```

        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')",
        "type": "com.sap.apptot.es.EventTypes"
    },
    {
        "Name": "core.automobiles:ExceededPressure",
        "PackageName": "core.automobiles",
        "PropertySetId": "WheelPressure",
        "PropertySetType": "core.automobiles:Wheel",
        "LanguageCode": "en",
        "Description": "Event type for exceeded pressure",
        "PropertySetDescriptions": {
            "__deferred": {
                "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')/PropertySetDescriptions"
            }
        },
        "Statuses": {
            "results": [{
                "EventStatus": "Open",
                "LanguageCode": "en",
                "Description": "Open Status"
            },
            {
                "EventStatus": "InProgress",
                "LanguageCode": "en",
                "Description": "In process Status"
            },
            {
                "EventStatus": "Completed",
                "LanguageCode": "en",
                "Description": "Completed Status"
            }
        ]
    },
    {
        "Severities": {
            "__deferred": {
                "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')/Severities"
            }
        },
        "Codes": {
            "__deferred": {
                "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')/Codes"
            }
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')/Descriptions"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
            "type": "com.sap.apptot.es.EventTypes"
        },
        "Name": "core.automobiles:ExceededTemperature",
        "PackageName": "core.automobiles",
        "PropertySetId": "EngineTemp",

```



```

        "PropertySetType": "core.automobiles:Engine",
        "LanguageCode": "en",
        "Description": "Event type for exceeded temperature",
        "PropertySetDescriptions": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/PropertySetDescriptions"
            }
        },
        "Statuses": {
            "results": [{
                "EventStatus": "InProgress",
                "LanguageCode": "en",
                "Description": "In process Status"
            },
            {
                "EventStatus": "Completed",
                "LanguageCode": "en",
                "Description": "Completed Status"
            },
            {
                "EventStatus": "Open",
                "LanguageCode": "en",
                "Description": "Open Status"
            }
        ]
    },
    "Severities": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Severities"
        }
    },
    "Codes": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Codes"
        }
    },
    "Descriptions": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Descriptions"
        }
    }
}
]
}
}

```

## Related Information

[Event Type \[page 520\]](#)

## 8.3.2.10 Read Severities of all Event Types

With this method, you can retrieve severities of all event types.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

### Request

**URI:** `/ES/v1/EventTypes?$expand=Severities`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

### Request Parameters

None

### Query Request Parameters

Parameter	Type	Required	Description
<code>\$top</code>	Integer	No	Number of records to include in the result set.
<code>\$skip</code>	Integer	No	Number of records to exclude from the result set.
<code>\$orderby</code>	String	No	Field name to be used for sorting the result set in ascending order; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> and <a href="#">EventSeverity</a> .  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.

Parameter	Type	Required	Description
\$select	String	No	Select condition for the data fields to be included in the result set; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> and <a href="#">EventSeverity</a> .
\$filter	String	No	Filter condition to be applied; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> and <a href="#">EventSeverity</a> .

## Request Example

```
/ES/v1/EventTypes?$expand=Severities
```

## Response

### Response Status and Error Codes

Code	Description
200	Severities of all event types are retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')",
        "type": "com.sap.apptot.es.EventTypes"
      },
      "Name": "core.automobiles:ExceededPressure",
      "PackageName": "core.automobiles",
      "PropertySetId": "WheelPressure",
      "PropertySetType": "core.automobiles:Wheel",
      "LanguageCode": "en",
      "Description": "Event type for exceeded pressure",
      "PropertySetDescriptions": {
        "__deferred": {
```

```

        "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededPressure')/PropertySetDescriptions"
    },
    "Statuses": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededPressure')/Statuses"
        }
    },
    "Severities": {
        "results": [{
            "EventSeverity": "1",
            "LanguageCode": "en",
            "Description": "High"
        },
        {
            "EventSeverity": "2",
            "LanguageCode": "en",
            "Description": "Medium"
        },
        {
            "EventSeverity": "3",
            "LanguageCode": "en",
            "Description": "Low"
        }
    ]
},
"Codes": {
    "__deferred": {
        "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededPressure')/Codes"
    }
},
"Descriptions": {
    "__deferred": {
        "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededPressure')/Descriptions"
    }
},
{
    "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
        "type": "com.sap.apptot.es.EventTypes"
    },
    "Name": "core.automobiles:ExceededTemperature",
    "PackageName": "core.automobiles",
    "PropertySetId": "EngineTemp",
    "PropertySetType": "core.automobiles:Engine",
    "LanguageCode": "en",
    "Description": "Event type for exceeded temperature",
    "PropertySetDescriptions": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/PropertySetDescriptions"
        }
    },
    "Statuses": {
        "__deferred": {

```

```

        "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Statuses"
    },
    "Severities": {
        "results": [{
            "EventSeverity": "1",
            "LanguageCode": "en",
            "Description": "High"
        },
        {
            "EventSeverity": "2",
            "LanguageCode": "en",
            "Description": "Medium"
        },
        {
            "EventSeverity": "3",
            "LanguageCode": "en",
            "Description": "Low"
        }
    ]
},
"Codes": {
    "__deferred": {
        "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Codes"
    }
},
"Descriptions": {
    "__deferred": {
        "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Descriptions"
    }
}
}
]
}
}

```

## Related Information

[Event Type \[page 520\]](#)

### 8.3.2.11 Read Codes of all Event Types

With this method, you can retrieve codes of all event types.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `ES/v1/EventTypes?$expand=Codes`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

## Request Parameters

None

## Query Request Parameters

Parameter	Type	Required	Description
<code>\$top</code>	Integer	No	Number of records to include in the result set.
<code>\$skip</code>	Integer	No	Number of records to exclude from the result set.
<code>\$orderby</code>	String	No	Field name to be used for sorting the result set in ascending order; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> and <a href="#">EventCode</a> .  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
<code>\$select</code>	String	No	Select condition for the data fields to be included in the result set; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> and <a href="#">EventCode</a> .
<code>\$filter</code>	String	No	Filter condition to be applied; valid fields are <a href="#">Name</a> , <a href="#">PackageName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> and <a href="#">EventCode</a> .

## Request Example

`ES/v1/EventTypes?$expand=Codes`

## Response

### Response Status and Error Codes

Code	Description
200	Codes of all event types retrieved successfully.

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')",
        "type": "com.sap.apptot.es.EventTypes"
      },
      "Name": "core.automobiles:ExceededPressure",
      "PackageName": "core.bbautomobiles",
      "PropertySetId": "WheelPressure",
      "PropertySetType": "core.automobiles:Wheel",
      "LanguageCode": "en",
      "Description": "Event type for exceeded pressure",
      "PropertySetDescriptions": {
        "__deferred": {
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')/PropertySetDescriptions"
        }
      },
      "Statuses": {
        "__deferred": {
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')/Statuses"
        }
      },
      "Severities": {
        "__deferred": {
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles%3AExceededPressure')/Severities"
        }
      },
      "Codes": {
        "results": [{
          "EventCode": "EQ12",
          "LanguageCode": "en",
          "Description": "Event Code 12"
        }]
      },
      "Descriptions": {
        "__deferred": {
```

```

        "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededPressure')/Descriptions"
    }
    },
    {
        "__metadata": {
            "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventTypes('core.automobiles%3AExceededTemperature')",
            "type": "com.sap.apptot.es.EventTypes"
        },
        "Name": "core.automobiles:ExceededTemperature",
        "PackageName": "core.automobiles",
        "PropertySetId": "EngineTemp",
        "PropertySetType": "core.automobiles:Engine",
        "LanguageCode": "en",
        "Description": "Event type for exceeded temperature",
        "PropertySetDescriptions": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/PropertySetDescriptions"
            }
        },
        "Statuses": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Statuses"
            }
        },
        "Severities": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Severities"
            }
        },
        "Codes": {
            "results": [{
                "EventCode": "EQ12",
                "LanguageCode": "en",
                "Description": "Event Code 12"
            },
            {
                "EventCode": "EQ14",
                "LanguageCode": "en",
                "Description": "Event Code 14"
            }
        ]
    },
    "Descriptions": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventTypes('core.automobiles
%3AExceededTemperature')/Descriptions"
        }
    }
}
]
}
}

```



## Related Information

[Event Type \[page 520\]](#)

### 8.3.3 Event Property Set Type

Event property set type is a group of attributes that are used to describe events

You can define event property set types to be assigned to an event. These event property set types belong to the data category [EventData](#) and are assigned to event types to describe an event. For example, **Exceeded Temperature** is a valid event property set type defined for an event type. **Temperature** of the engine and the message displayed when the temperature exceeds are some of the properties that you can define for the corresponding event property set type.

The following are the supported predefined annotations for an event property set type:

- `com.sap.apptot.security:pii`
  - You can assign this annotation to an event property set type that contains data to identify a person or subject. Data such as nick name, birthdate, and email address are few examples that can be categorized as data that identifies a person.
- `com.sap.apptot.security:spi`
  - You can assign this annotation to an event property set type that contains critical data of a person. Data such as identification number, ethnic origin, and biometric data are few examples that can be categorized as critical data of a person.

**Resource Path:** `/ES/v1/EventPropertySetTypes`

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scope
<i>POST</i>	<a href="#">Create Event Property Set Type [page 559]</a>	<code>/ES/v1/EventPropertySetTypes</code>	<code>&lt;tde&gt;.c</code>
<i>GET</i>	<a href="#">Read Details of an Event Property Set Type [page 562]</a>	<code>/ES/v1/ EventPropertySetTypes('&lt;event property set type name&gt;')</code>	<code>&lt;tde&gt;.r</code>
<i>GET</i>	<a href="#">Read all Event Property Set Types [page 567]</a>	<code>/ES/v1/EventPropertySetTypes</code>	<code>&lt;tde&gt;.r</code>
<i>PATCH</i>	<a href="#">Update an Event Property Set Type [page 574]</a>	<code>/ES/v1/ EventPropertySetTypes('&lt;event property set type name&gt;')</code>	<code>&lt;tde&gt;.u</code>

HTTP Method	Operation	URI	Scope
DELETE	<a href="#">Delete an Event Property Set Type</a> <a href="#">[page 576]</a>	/ES/v1/ EventPropertySetTypes('<event property set type name>')	<tde>.d

## Event Property Set Type Object Properties

Property	Type	Mandatory	Maximum Length	Description
Name	String	Yes	81	Name of the event type property set type
Description	String	No	255	Description of the event property set type
PackageName	String	Yes	50	Name of the package for which the event types are configured
DataCategory	Enum	Yes	-	Data category of the property set type
<div> <div>i Note</div> <div>Only allowed value is <a href="#">EventData</a>.</div> </div>				
Properties	Array of properties	No	-	Array of properties of data category EventData

## Event Properties Object Properties

Property	Type	Mandatory	Maximum Length	Description
Name	String	Yes	30	Name of the property
Description	String	No	255	Description of the property
PropertySetType	String	Yes	81	Name of the property set type
Type	enum	Yes	30	Type used by the property
PropertyLength	Integer	Yes	30	Storage length for the property type specified as an integer value

## Description Properties

Property	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the object description
Description	String	255	Description of the object

## Related Information

[Event Type \[page 520\]](#)

[Data Protection and Privacy \[page 322\]](#)

### 8.3.3.1 Create Event Property Set Type

With this method, you can create an event property set type in a specific package.

#### Naming convention for property set type

- Always prefixed with the package name.
- Can be a combination of alphanumeric characters and underscore.
- Each name must be fully qualified and contains a package name, colon as a separator and the object name. Maximum length of fully qualified name is 81 characters.
- Do not start the object name with a digit or an underscore.
- Length should be at least 3 characters.
- Camel case is generally allowed, however there is a uniqueness check carried out against textual case conflicts:
  - Two names with same characters of different textual case is not allowed. For example: A value "AbcDef" is not accepted when there is another value "abcDEF" that exists.
- Verify that a colon is used only to separate a package name and the object name. The syntax is: `<package>:<name>`. For example: `core.automobiles:FlyWheel`
- Name must be unique within a package.

#### Naming convention for properties

- Only the following characters are supported:
  - Uppercase and lowercase alphabets a through z and A through Z
  - Numeric digits 0 through 9
  - Punctuation marks underscore (\_), hyphen (-), colon (:), and full-stop (.)
- Maximum length is 30 characters.
- Property names have to be unique for a given set of property set types.

## Request

**URI:** /ES/v1/EventPropertySetTypes

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** <td>.c

### Request Headers

Header	Required	Values
Accept	Yes	application/json

### Request Parameters

None

### Request Example

```
/ES/v1/EventPropertySetTypes
```

## Payload

The following request payload creates event property set type `core.automobiles:Engine` in the package `core.automobiles`.

```
{
  "Name": "core.automobiles:Engine",
  "PackageName": "core.automobiles",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Engine"
  },
  {
    "LanguageCode": "de",
    "Description": "Motor"
  }
],
  "DataCategory": "EventData",
  "Properties": [{
    "Name": "Temperature",
    "Type": "String",
    "PropertyLength": "127",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Engine Temperature"
    },
    {
      "LanguageCode": "de",
      "Description": "Motortemperatur"
    }
  ]
}]
}
```

## Response

### Response Status and Error Codes

Code	Description
200	Event property set type created successfully.

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples.

### Response Payload Example

The following is the response payload after creating the event property set type `core.automobiles:Engine`.

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
          "type": "com.sap.apptot.es.EventPropertySetTypes"
        },
        "Name": "core.automobiles:Engine",
        "PackageName": "core.automobiles",
        "DataCategory": "EventData",
        "Language": null,
        "PropertySetTypeDescription": null,
        "Properties": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')/Properties"
          }
        },
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')/Descriptions"
          }
        }
      }
    ]
  }
}
```

## Related Information

[Event Property Set Type \[page 557\]](#)

### 8.3.3.2 Read Details of an Event Property Set Type

Retrieves details of an event property set type with the specified name from the database

With this method, you send a request to the server to retrieve a particular event property set type specified by its name.

#### Request

**URI:** `/ES/v1/EventPropertySetTypes('<event property set type name>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

#### Request Headers

Header	Required	Values
Accept	Yes	application/json

#### Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	String	Unique name of the event property set type

#### Request Example

`/ES/v1/EventPropertySetTypes('core.automobiles:Engine')`

#### Response

#### Response Status and Error Codes

Code	Description
200	Details of a specific event property set type retrieved successfully.

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

### Example 1

`/ES/v1/EventPropertySetTypes('core.automobiles:Engine')`

Retrieves details of the specified event property set type

```
{
  "d": {
    "__metadata": {
      "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
      "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
      "type": "com.sap.apptot.es.EventPropertySetTypes"
    },
    "Name": "core.automobiles:Engine",
    "PackageName": "core.automobiles",
    "DataCategory": "EventData",
    "Language": "en",
    "PropertySetTypeDescription": "Engine",
    "Properties": {
      "__deferred": {
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')/Properties"
      }
    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')/Descriptions"
      }
    }
  }
}
```

### Example 2

`/ES/v1/EventPropertySetTypes('core.automobiles:Engine')?$expand=Properties`

Retrieves details of the specified event property set type along with the detailed list of properties

```
{
  "d": {
    "__metadata": {
      "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
      "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
      "type": "com.sap.apptot.es.EventPropertySetTypes"
    },
    "Name": "core.automobiles:Engine",
    "PackageName": "core.automobiles",
    "DataCategory": "EventData",
    "Language": "",
    "PropertySetTypeDescription": "",
    "Properties": {
      "results": [
        {
          "Name": "Speed",

```

```

        "PropertySetType": "core.automobiles:Engine",
        "Type": "Numeric",
        "PropertyLength": null
      },
      {
        "Name": "Temperature",
        "PropertySetType": "core.automobiles:Engine",
        "Type": "String",
        "PropertyLength": "128"
      }
    ]
  },
  "Descriptions": {
    "__deferred": {
      "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')/Descriptions"
    }
  }
}

```

### Example 3

`/ES/v1/EventPropertySetTypes('core.automobiles:Engine')?$expand=Descriptions`

Retrieves details of the specified event property set type along with the detailed list of descriptions of event property set types

```

{
  "d": {
    "__metadata": {
      "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
      "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
      "type": "com.sap.apptot.es.EventPropertySetTypes"
    },
    "Name": "core.automobiles:Engine",
    "PackageName": "core.automobiles",
    "DataCategory": "EventData",
    "Language": "",
    "PropertySetTypeDescription": "",
    "Properties": {
      "__deferred": {
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')/Properties"
      }
    },
    "Descriptions": {
      "results": [
        {
          "LanguageCode": "en",
          "Description": "Engine"
        },
        {
          "LanguageCode": "de",
          "Description": "Motor"
        }
      ]
    }
  }
}

```

### Example 4

`/ES/v1/EventPropertySetTypes('core.automobiles:Engine')/Properties`



Retrieves the list of properties for the specified event property set type

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Properties(PropertyName='Speed',PropertySetType='core.automobiles%3AEngine')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Properties(PropertyName='Speed',PropertySetType='core.bbautomobiles%3AEngine')",
          "type": "com.sap.apptot.es.Properties"
        },
        "Name": "Speed",
        "PropertySetType": "core.automobiles:Engine",
        "Type": "Numeric",
        "PropertyLength": "",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Properties(PropertyName='Speed',PropertySetType='core.automobiles%3AEngine')/Descriptions"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Properties(PropertyName='Temperature',PropertySetType='core.automobiles%3AEngine')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Properties(PropertyName='Temperature',PropertySetType='core.automobiles%3AEngine')",
          "type": "com.sap.apptot.es.Properties"
        },
        "Name": "Temperature",
        "PropertySetType": "core.automobiles:Engine",
        "Type": "String",
        "PropertyLength": "128",
        "Descriptions": {
          "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Properties(PropertyName='Temperature',PropertySetType='core.automobiles%3AEngine')/Descriptions"
          }
        }
      }
    ]
  }
}
```

#### Example 5

/ES/v1/EventPropertySetTypes('core.automobiles:Engine')/Descriptions

Retrieves the list of descriptions for the specified event property set type

```
{
  "d": {
    "results": [
      {
        "__metadata": {
```

```

        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('en')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('en')",
        "type": "com.sap.apptot.es.Descriptions"
    },
    "LanguageCode": "en",
    "Description": "Engine"
},
{
    "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('de')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Descriptions('de')",
        "type": "com.sap.apptot.es.Descriptions"
    },
    "LanguageCode": "de",
    "Description": "Motor"
}
]
}
}

```

#### Example 6

`/ES/v1/EventPropertySetTypes('core.automobiles:Engine')?$expand=Properties/Descriptions`

Retrieves details of the specified event property set type along with the expanded list of properties and descriptions of the event property set type

```

{
  "d": {
    "__metadata": {
      "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
      "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
      "type": "com.sap.apptot.es.EventPropertySetTypes"
    },
    "Name": "core.automobiles:Engine",
    "PackageName": "core.automobiles",
    "DataCategory": "EventData",
    "Language": "",
    "PropertySetTypeDescription": "",
    "Properties": {
      "results": [{
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Properties(PropertyName='Speed',PropertySetType='core.automobiles%3AEngine')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Properties(PropertyName='Speed',PropertySetType='core.automobiles%3AEngine')",
          "type": "com.sap.apptot.es.Properties"
        },
        "Name": "Speed",
        "PropertySetType": "core.automobiles:Engine",
        "Type": "Numeric",
        "PropertyLength": null,
        "Descriptions": {
          "results": [{
            "LanguageCode": "de",
            "Description": "Motor"
          }
        ]
      }
    ]
  }
}

```

```

        {
            "LanguageCode": "en",
            "Description": "Engine Speed"
        }
    ]
},
{
    "__metadata": {
        "id": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/
Properties(PropertyName='Temperature',PropertySetType='core.automobiles
%3AEngine')",
        "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/
Properties(PropertyName='Temperature',PropertySetType='core.automobiles
%3AEngine')",
        "type": "com.sap.apptot.es.Properties"
    },
    "Name": "Temperature",
    "PropertySetType": "core.automobiles:Engine",
    "Type": "String",
    "PropertyLength": "128",
    "Descriptions": {
        "results": [{
            "LanguageCode": "en",
            "Description": "Engine Temperature"
        },
        {
            "LanguageCode": "de",
            "Description": "Motortemperatur"
        }
    ]
}
]
},
{
    "Descriptions": {
        "__deferred": {
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')/Descriptions"
        }
    }
}
}
}

```

## Related Information

[Event Property Set Type \[page 557\]](#)

### 8.3.3.3 Read all Event Property Set Types

Retrieves a list of event property set types from the database

With this method, you send a request to the server to retrieve all event property set types or a subset of event property set types according to the filter criteria provided. For a collective request for a set of event property set types, you have these options:

- You can restrict the set of event property set types to be retrieved by filter criteria based on name, version, package, and data category.
- You can define that the set of matching event property set types shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ES/v1/EventPropertySetTypes`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

### Request Headers

Header	Required	Values
Accept	Yes	application/json

### Request Parameters

None

### Query Request Parameters

Parameter	Type	Required	Description
<code>\$top</code>	Integer	No	Number of records to include in the result set.
<code>\$skip</code>	Integer	No	Number of records to exclude from the result set.
<code>\$orderby</code>	String	No	Field name to be used for sorting the result set in ascending order; valid fields are <a href="#">Name</a> , <a href="#">Version</a> , <a href="#">PackageName</a> , and <a href="#">DataCategory</a> .  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.

Parameter	Type	Required	Description
\$select	String	No	Select condition for the data fields to be included in the result set; valid fields are <a href="#">Name</a> , <a href="#">Version</a> , <a href="#">PackageName</a> , and <a href="#">DataCategory</a> .
\$filter	String	No	Filter condition to be applied; valid fields are <a href="#">Name</a> , <a href="#">Version</a> , <a href="#">PackageName</a> , and <a href="#">DataCategory</a> .

## Request Example

```
/ES/v1/EventPropertySetTypes
```

## Response

### Response Status and Error Codes

Code	Description
200	Event property set types retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as depicted in the following examples:

### Example 1

```
/ES/v1/EventPropertySetTypes
```

Retrieves all event property set types

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
          "type": "com.sap.apptot.es.EventPropertySetTypes"
        },
        "Name": "core.automobiles:Engine",
        "PackageName": "core.automobiles",
        "DataCategory": "EventData",
      }
    ]
  }
}
```

```

        "Language": "en",
        "PropertySetTypeDescription": "Engine",
        "Properties": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AEngine')/Properties"
            }
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AEngine')/Descriptions"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AWheel')",
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AWheel')",
            "type": "com.sap.apptot.es.EventPropertySetTypes"
        },
        "Name": "core.automobiles:Wheel",
        "PackageName": "core.bbautomobiles",
        "DataCategory": "EventData",
        "Language": "en",
        "PropertySetTypeDescription": "Wheel",
        "Properties": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AWheel')/Properties"
            }
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AWheel')/Descriptions"
            }
        }
    }
]
}
}

```

## Example 2

/ES/v1/EventPropertySetTypes?\$expand=Properties

Retrieves all event property set types with the expanded list of their properties

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
          "type": "com.sap.apptot.es.EventPropertySetTypes"
        },
        "Name": "core.automobiles:Engine",

```

```

        "PackageName": "core.automobiles",
        "DataCategory": "EventData",
        "Language": "en",
        "PropertySetTypeDescription": "Engine Speed",
        "Properties": {
            "results": [
                {
                    "Name": "Speed",
                    "PropertySetType": "core.automobiles:Engine",
                    "Type": "Numeric",
                    "PropertyLength": null
                }
            ]
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AEngine')/Descriptions"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
            "type": "com.sap.apptot.es.EventPropertySetTypes"
        },
        "Name": "core.automobiles:Engine",
        "PackageName": "core.automobiles",
        "DataCategory": "EventData",
        "Language": "en",
        "PropertySetTypeDescription": "Engine Temperature",
        "Properties": {
            "results": [
                {
                    "Name": "Temperature",
                    "PropertySetType": "core.automobiles:Engine",
                    "Type": "String",
                    "PropertyLength": "128"
                }
            ]
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AEngine')/Descriptions"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AWheel')",
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AWheel')",
            "type": "com.sap.apptot.es.EventPropertySetTypes"
        },
        "Name": "core.automobiles:Wheel",
        "PackageName": "core.automobiles",
        "DataCategory": "EventData",
        "Language": "en",
        "PropertySetTypeDescription": "Wheel Temperature",
        "Properties": {
            "results": [

```

```

        {
            "Name": "Temperature",
            "PropertySetType": "core.automobiles:Wheel",
            "Type": "String",
            "PropertyLength": "128"
        }
    ],
    },
    "Descriptions": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AWheel')/Descriptions"
        }
    },
    {
        "__metadata": {
            "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AWheel')",
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AWheel')",
            "type": "com.sap.apptot.es.EventPropertySetTypes"
        },
        "Name": "core.automobiles:Wheel",
        "PackageName": "core.automobiles",
        "DataCategory": "EventData",
        "Language": "en",
        "PropertySetTypeDescription": "Wheel Pressure",
        "Properties": {
            "results": [
                {
                    "Name": "Pressure",
                    "PropertySetType": "core.automobiles:Wheel",
                    "Type": "String",
                    "PropertyLength": "127"
                }
            ]
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AWheel')/Descriptions"
            }
        }
    }
]
}
}

```

### Example 3

/ES/v1/EventPropertySetTypes?\$expand=Descriptions

Retrieves all event property set types with the expanded list of their descriptions

```

{
    "d": {
        "results": [{
            "__metadata": {
                "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
                "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AEngine')",
                "type": "com.sap.apptot.es.EventPropertySetTypes"
            },

```



```

        "Name": "core.automobiles:Engine",
        "PackageName": "core.automobiles",
        "DataCategory": "EventData",
        "Language": "",
        "PropertySetTypeDescription": "",
        "Properties": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AEngine')/Properties"
            }
        },
        "Descriptions": {
            "results": [{
                "LanguageCode": "de",
                "Description": "Motor"
            },
            {
                "LanguageCode": "en",
                "Description": "Engine"
            }
        ]
    },
    "Annotations": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AEngine')/Annotations"
        }
    }
},
{
    "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AWheel')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventPropertySetTypes('core.automobiles%3AWheel')",
        "type": "com.sap.apptot.es.EventPropertySetTypes"
    },
    "Name": "core.automobiles%3AWheel",
    "PackageName": "core.automobiles",
    "DataCategory": "EventData",
    "Language": "",
    "PropertySetTypeDescription": "",
    "Properties": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AWheel')/Properties"
        }
    },
    "Descriptions": {
        "results": [{
            "LanguageCode": "en",
            "Description": "Wheel"
        }
    ]
    },
    "Annotations": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventPropertySetTypes('core.automobiles
%3AWheel')/Annotations"
        }
    }
}
]
}
}

```

## Related Information

[Event Property Set Type \[page 557\]](#)

### 8.3.3.4 Update an Event Property Set Type

With this method, you can to update the specified event property set type.

The *ETag* value is mandatory to update an event property set type. You must retrieve the *ETag* value of the event property set type using the `GET (/ES/v1/EventPropertySetTypes('event property set type name'))` service. The *ETag* value is assigned to the request header parameter `If-Match` in the `PATCH` request. If you do not include the *ETag* value in the request header or if the *ETag* value is not correct, the server displays an error message.

#### i Note

- You must specify the package name (to which the event property set type belongs) in the request payload.
- You cannot change the name of the event property set type using `PATCH` request.
- The valid operations for the properties are `ADD`, `DELETE`, and `UPDATE`.
  - You can add new properties, edit attributes of existing properties, or delete existing properties. You can edit and delete properties only if the event data does not exist.
- The valid operations for the annotations are `ADD`, `DELETE`, and `UPDATE`.
  - You can add annotation for an event property set type if nothing is already assigned and even though the data exists for the properties
  - You cannot add multiple annotations for the event property set type
  - You cannot add, delete, or change the annotation for an event property set type if it is referred in another event property set type
  - You cannot delete or change the already assigned annotation, if event data exists for the properties of the event property set type

After updating the event property set type, the system generates a new *ETag* value for the event property set type and replaces the old value with the new value.

## Request

**URI:** `/ES/v1/EventPropertySetTypes('<event property set type name>')`

**Operation Type:** `CRUD`

**HTTP Method:** *PATCH*

**Permissions:** `<td>.u`

## Request Headers

Header	Required	Values
If-Match	Yes	Latest <a href="#">ETag</a> value of a specific event property set type that is updated using the PATCH request.

## Request Parameters

Parameter	Required	Description
Name	Yes	Unique name of the event property set type

## Request Example

```
/ES/v1/EventPropertySetTypes('core.automobiles:Engine')
```

The following request payload adds a new property "Speed" for the property set type `core.automobiles:Engine`:

```
{
  "PackageName": "core.automobiles",
  "Properties": [{
    "Name": "Speed",
    "Operation": "ADD",
    "Type": "Numeric",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Engine Speed"
    },
    {
      "LanguageCode": "de",
      "Description": "Motordrehzahl"
    }
  ]
}]
}
```

## Response

### Response Status and Error Codes

Code	Description
204	Event property set type updated successfully.

## Related Information

[Event Property Set Type \[page 557\]](#)

## 8.3.3.5 Delete an Event Property Set Type

With this method, you can delete an event property set type with a specific name.

The *ETag* value is mandatory to delete an event property set type. You must retrieve the *ETag* value of the event property set type using the GET (/ES/v1/EventPropertySetTypes('<event property set type name>')) service. The *ETag* value is assigned to the request header parameter If-Match in the PATCH request. If you do not include the *ETag* value in the request header or if the *ETag* value is not correct, the server displays an error message.

### Request

**URI:** /ES/v1/EventPropertySetTypes('<event property set type name>')

**Operation Type:** CRUD

**HTTP Method:** GET

**Permissions:** <tded>.r

### Request Parameters

None

### Request Example

```
/ES/v1/EventPropertySetTypes('core.automobiles:Engine')
```

### Response

#### Response Status and Error Codes

Code	Description
204	Data deleted successfully.

### Related Information

[Event Property Set Type \[page 557\]](#)

## 8.4 Thing Type and Sensor Type Mapping

Service for managing the mapping of a thing type to a sensor type

### Pre-requisites

Configure the IoT service credentials using Thing Modeler data mapping credentials API:

<https://tm-data-mapping.cfapps.sap.hana.ondemand.com/v1/credentials>

See [Create Credentials \[page 754\]](#).

### Functional Overview

Thing Type OData service supports auto creation of Capabilities, Sensor Types, Property Set Types, Property Sets, Thing Type and Mappings in a single API. This is possible by using the following enhancements of the Thing Type OData service. See [Thing Type \[page 368\]](#).

- New entities such as Capability, Sensor Type and Sensor Type to Thing Type Mapping
- An association between Thing Type Entity and Sensor Types
- An association between Thing Type and Sensor Type Mapping
- The above enhancement facilitates to auto create all the required artifacts in a single API call as per the OData specifications.
- This API will internally perform the following actions to create the artifacts:
  - REST APIs of IoT Services will be used to create Capabilities, Sensor Types and obtain the respective object IDs.
  - Thing Type is created in SAP IoT AE.
  - Sensor Type to Thing Type Mapping is created in SAP IoT AE.
- These enhancements are fully backward compatible and do not impact the existing usage of these APIs.
- If the additional data structures are found in the payload, then these API's will take care of auto creation.

**Resource Path:** `v2/Packages('<package name>')/ThingTypes`

**Example for a base URI in a cloud foundry environment:** <https://sap-iotaeexplore.iot-sap.cfapps.eu10.hana.ondemand.com/config-thing-sap/ThingConfiguration>

### Operations

#### CRUD Operations

HTTP Method	Operation	URI	Scope
<i>POST</i>	<a href="#">Create a Thing Type and Sensor Type Mapping [page 581]</a>	<code>/ThingConfiguration/v2/ Packages('&lt;package name&gt;')/ ThingTypes</code>	<code>&lt;thingconf &gt;.c</code>

HTTP Method	Operation	URI	Scope
GET	<a href="#">Read a Thing Type with its Sensor Type Mappings [page 585]</a>	/ThingConfiguration/v2/ ThingTypes('<thing type name>')? \$expand=SensorTypeMappings/ MeasureMappings,SensorTypeMappings /TargetMappings	<thingconf >.r
GET	<a href="#">Read a Thing Type with its Sensor Types used in Mappings [page 588]</a>	ThingConfiguration/v2/ ThingTypes('<thing type name>')? \$expand=SensorTypes/ SensorTypeCapabilities/ Capabilities/Properties	<thingconf >.r
PATCH	<a href="#">Update Sensor Type Mappings or Sensor Types in the Thing Type [page 593]</a>	/ThingConfiguration/v2/ ThingTypes(<Thing Type Name>)	<thingconf >.r
DELETE	<a href="#">Delete a Thing Type with its Sensor Type Mappings [page 596]</a>	/ThingConfiguration/v2/ ThingTypes(<Thing Type Name>)	<thingconf >.d

#### i Note

- <thingconf> refers to the thing configuration application name.
- <version> refers to the version of the method, /v2

## Thing Type Object Properties

Property	Type	Mandatory	Maximum Length	Description
Name	String	Yes	81	Unique name of the thing type
Description	String	No	60	Description of the thing type for each language with the corresponding ISO language code
PropertySets	Array of property set objects	No	-	Array of property sets that belong to the thing type

#### i Note

If you include `Descriptions` along with the `CopySource` field, the method assigns these descriptions for the newly created thing type. If you do not include `Descriptions` along with the `CopySource` field, the method copies descriptions defined in the source thing type to the newly created thing type.

Property	Type	Mandatory	Maximum Length	Description
ThingTypeCategory	String	No	Fixed	<p>Category of thing type</p> <p>Only the following values are supported:</p> <ul style="list-style-type: none"> <li><a href="#">ASSET</a> – Indicates the root thing</li> <li><a href="#">COMPONENT</a> – Indicates the sub-items of the <a href="#">ASSET</a></li> </ul> <div> <p><b>i Note</b></p> <p>You can define this only using /v2 version of the create method.</p> </div>
CopySource	String	No	81	<p>Name of the thing type</p> <p>You can use this field to specify the already existing thing type name, structure of which is copied to the new thing type to be created.</p> <div> <p><b>i Note</b></p> <ul style="list-style-type: none"> <li>You can use this field only in the /v1 version of the create method.</li> <li>You cannot specify any value other than the thing type name in this field.</li> <li>You cannot create copy of a thing type that belongs to a package with scope=private.</li> <li>In a request payload to copy an existing thing type, you can only specify the name of the new thing type, CopySource field, and Descriptions field.</li> </ul> </div>

## SensorTypeMappings Properties

Property	Type	Mandatory	Maximum Length	Description
MappingId	String	No	81	ID of the mapping
Name	String	Yes	81	Name of the mapping
Description	String	No	60	Description of the mapping
MeasureMappings	Array	No	-	Array of MeasureMappings

Property	Type	Mandatory	Maximum Length	Description
TargetMappings	Array	No	-	Array of TargetMappings

### MeasureMappings Properties

Property	Type	Mandatory	Maximum Length	Description
MappingId	String	No	81	ID of its parent mapping
SensorTypeName	String	Yes	81	Name of sensor type
SensorTypeId	String	No	81	ID of sensor type
CapabilityName	String	Yes	60	Name of capability
CapabilityId	String	No	60	ID of capability
CapabilityPropertyId	String	Yes	81	Name of Capability Property
ThingTypeName	String	Yes	81	Name of thing type
PropertySetTypeId	String	Yes	81	ID of Property Set Type
NamedPropertySetTypeId	String	Yes	81	ID of Named Property Set Type
NamedPropertySetTypePropertyId	String	Yes	81	ID of Property Set Type Property

### TargetMappings Properties

Property	Type	Mandatory	Maximum Length	Description
MappingId	String	No	81	ID of its parent mapping
SensorTypeName	String	Yes	81	Name of sensor type
SensorTypeId	String	No	81	ID of sensor type
CapabilityName	String	Yes	60	Name of capability
CapabilityId	String	No	60	ID of capability
CapabilityPropertyId	String	Yes	81	Name of Capability Property
ThingTypeName	String	Yes	81	Name of thing type
PropertySetTypeId	String	Yes	81	ID of Reference Property Set Type
NamedPropertySetTypeId	String	Yes	81	ID of Named Property Set Type
NamedPropertySetTypeReferencePropertyId	String	Yes	81	ID of Reference Property Set Type Property of attribute type either <code>com.sap.iot.core.TargetValue</code> or <code>com.sap.iot.core.Command</code>



### SensorTypes Properties

Property	Type	Mandatory	Maximum Length	Description
id	String	No	81	ID of Sensor Type
Name	String	Yes	81	Name of Sensor Type
SensorTypeCapabilities	Array	Yes	-	Array of SensorTypeCapabilities

### SensorTypeCapabilities Properties

Property	Type	Mandatory	Maximum Length	Description
Capabilities	Object	Yes	-	Capability Object
Type	String	Yes	81	Type of capability. It can be either <code>measure</code> or <code>command</code> .

### Capabilities Properties

Property	Type	Mandatory	Maximum Length	Description
id	String	No	81	ID of Capability
Name	String	Yes	81	Name of Capability
Properties	Array	-	-	Array of Properties

### Properties

Property	Type	Mandatory	Maximum Length	Description
Name	String	Yes	81	Name of Capability Property
DataType	String	Yes	81	Data Type of Capability Property
UnitOfMeasure	String	No	50	Unit of Measure of Capability Property

### Thing Type Description Properties

Property	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the object description
Description	String	255	Description of the object

## 8.4.1 Create a Thing Type and Sensor Type Mapping

With this method, you create thing types and sensor type mappings in a specific package.

### Naming convention for thing type

- Always prefixed with the package name.
- Can be a combination of alphanumeric characters and underscore.

- Each name must be fully qualified and contains a package name, colon as a separator and the object name. Maximum length of fully qualified name is 81 characters.
- Do not start the object name with a digit or an underscore.
- Length should be at least 3 characters.
- Camel case is generally allowed, however there is a uniqueness check carried out against textual case conflicts:
  - Two names with same characters of different textual case is not allowed. For example: A value "AbcDef" is not accepted when there is another value "abcDEF" that exists.
- Verify that a colon is used only to separate a package name and the object name. The syntax is: <package>:<name>. For example: core.automobiles:FlyWheel
- Name must be unique within a package.

## Request

**URI:** /ThingConfiguration/v2/Packages('<package name>')/ThingTypes

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** <thingconf>.c

### Request Headers

Header	Required	Values
Accept-Language	No	Language of the thing type description. The default language is EN.

### Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	string	Unique name of the package

### Request Example

```
/ThingConfiguration/v2/
Packages('iotae.sandboxhdibd1.conv.services.demo.package')/ThingTypes
```

Creates the thing type iotae.sandboxhdibd1.conv.services.demo.package:ConvCoolerType in the package iotae.sandboxhdibd1.conv.services.demo.package. This also creates sensor types and capabilities in the IoT services, and sensor type mappings in the IoT AE.

#### **i** Note

Sensor type, capability and property names in sensor types JSON fragment should match with sensor type, capability and property names in Sensor Type Mapping JSON fragment.

## Payload

Format: *JSON*

```
{
  "Name": "iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "DemoCooler"
  }],
  "PropertySets": [{
    "Name": "TemperatureNPST",
    "PropertySetType":
      "iotae.sandboxhdibd1.conv.services.demo.package:TemperaturePST",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Temperature  properties"
    }]
  }],
  {
    "Name": "DoorOpenStatusNPST",
    "PropertySetType":
      "iotae.sandboxhdibd1.conv.services.demo.package:DoorOpenStatusPST",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Door Open Status  properties"
    }]
  },
  {
    "Name": "CompressorNPST",
    "PropertySetType":
      "iotae.sandboxhdibd1.conv.services.demo.package:CompressorPST",
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Compressor  properties"
    }]
  }],
  "SensorTypeMappings": [{
    "Descriptions": [{
      "LanguageCode": "en",
      "Description": "Sensor Type Mappings"
    }],
    "Name": "DemoDemoCoolerSensorTypeMappings",
    "MeasureMappings": [{
      "SensorTypeName": "DemoReedSensorType",
      "CapabilityName": "DemoDoorOpenStatusCapability",
      "CapabilityPropertyId": "DemoDoorOpenStatus",
      "PropertySetTypeId":
        "iotae.sandboxhdibd1.conv.services.demo.package:DoorOpenStatusPST",
      "NamedPropertySetTypeId": "DoorOpenStatusNPST",
      "NamedPropertySetTypePropertyId": "DoorOpenStatus"
    }],
    {
      "SensorTypeName": "DemoCoolerSensorType",
      "CapabilityName": "DemoTemperatureCapability",
      "CapabilityPropertyId": "DemoHumidity",
      "PropertySetTypeId":
        "iotae.sandboxhdibd1.conv.services.demo.package:TemperaturePST",
      "NamedPropertySetTypeId": "TemperatureNPST",
      "NamedPropertySetTypePropertyId": "HumidityProp"
    }],
    "TargetMappings": [{
      "SensorTypeName": "DemoPressureSensorType",
      "CapabilityName": "DemoPressureCapability",
      "CapabilityPropertyId": "DemoPressure",
      "PropertySetTypeId":
        "iotae.sandboxhdibd1.conv.services.demo.package:ReferencePST",
```

```

"NamedPropertySetTypeId": "CompressorNPST",
"NamedPropertySetTypeReferencePropertyId": "PressureTargetValue"
},
{
  "SensorTypeName": "DemoPressureSensorType",
  "CapabilityName": "DemoPressureCapability",
  "CapabilityPropertyId": "DemoRotSpeed",
  "PropertySetTypeId":
    "iotae.sandboxhdbdl.conv.services.demo.package:ReferencePST",
  "NamedPropertySetTypeId": "CompressorNPST",
  "NamedPropertySetTypeReferencePropertyId": "rotSpeedTargetValue"
}]
}],
"SensorTypes": [{
  "Name": "DemoCoolerSensorType",
  "SensorTypeCapabilities": [{
    "Capability": {
      "Name": "DemoTemperatureCapability",
      "Properties": [{
        "Name": "DemoTemperature",
        "DataType": "Double",
        "UnitOfMeasure": "C"
      }],
      {
        "Name": "DemoHumidity",
        "DataType": "Double",
        "UnitOfMeasure": "NM"
      }
    ]
  }],
  "Type": "measure"
}],
{
  "Name": "DemoReedSensorType",
  "SensorTypeCapabilities": [{
    "Capability": {
      "Name": "DemoDoorOpenStatusCapability",
      "Properties": [{
        "Name": "DemoDoorOpenStatus",
        "DataType": "Double",
        "UnitOfMeasure": "C"
      }],
      {
        "Name": "DemoPressureSensorType",
        "SensorTypeCapabilities": [{
          "Capability": {
            "Name": "DemoPressureCapability",
            "Properties": [{
              "Name": "DemoPressure",
              "DataType": "Double",
              "UnitOfMeasure": "C"
            }],
            {
              "Name": "DemoRotSpeed",
              "DataType": "Double",
              "UnitOfMeasure": "NM"
            }
          ]
        }],
        "Type": "command"
      }
    ]
  }],
  {
    "Name": "DemoPressureSensorType",
    "SensorTypeCapabilities": [{
      "Capability": {
        "Name": "DemoPressureCapability",
        "Properties": [{
          "Name": "DemoPressure",
          "DataType": "Double",
          "UnitOfMeasure": "C"
        }],
        {
          "Name": "DemoRotSpeed",
          "DataType": "Double",
          "UnitOfMeasure": "NM"
        }
      ]
    }],
    "Type": "command"
  }
]
}

```

## Response

### Response Status and Error Codes

Code	Description
201	Thing type and sensor type mapping created successfully

## 8.4.2 Read a Thing Type with its Sensor Type Mappings

With this method, you can retrieve the details of a thing type with its sensor type mappings.

### Request

**URI:** `/ThingConfiguration/v2/ThingTypes('<thing type name>')?$expand=SensorTypeMappings/MeasureMappings,SensorTypeMappings/TargetMappings`

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** `<thingconf>.r`

### Request Headers

Header	Required	Values
Accept-Language	No	Language of the thing type description. The default language is <i>en</i> .

### Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	string	Name of the thing type

### Request Example

`ThingConfiguration/v2/ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType')?$expand=SensorTypeMappings/MeasureMappings,SensorTypeMappings/TargetMappings`

```
{
  "d": {
    "__metadata": {
      "id": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package%3ADemoCoolerType')",
      "uri": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package%3ADemoCoolerType')",
      "type": "com.sap.apptot.ThingType"
    }
  }
}
```

```

    },
    "Name": "iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType",
    "PackageName": "iotae.sandboxhdibd1.conv.services.demo.package",
    "Description": "DemoCooler",
    "ExtendedThingType": null,
    "ThingTypeCategory": null,
    "PropertySets": {
      "__deferred": {
        "uri": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v2/
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package%3ADemoCoolerType')/
PropertySets"
      }
    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v2/
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package%3ADemoCoolerType')/
Descriptions"
      }
    },
    "ThingTemplates": {
      "__deferred": {
        "uri": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v2/
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package%3ADemoCoolerType')/
ThingTemplates"
      }
    },
    "SensorTypeMappings": {
      "results": [
        {
          "__metadata": {
            "id": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypeMappings('b807fece-b5b9-44ed-a7b4-4daffba5a691')",
            "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypeMappings('b807fece-b5b9-44ed-a7b4-4daffba5a691')",
            "type": "com.sap.apptot.SensorTypeMapping"
          },
          "MappingId": "b807fece-b5b9-44ed-a7b4-4daffba5a691",
          "Name": "DemoDemoCoolerSensorTypeMappings",
          "Description": null,
          "ThingTypeName":
"iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType",
          "Descriptions": {
            "__deferred": {
              "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypeMappings('b807fece-b5b9-44ed-a7b4-4daffba5a691')/Descriptions"
            }
          },
          "ThingTypes": {
            "__deferred": {
              "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypeMappings('b807fece-b5b9-44ed-a7b4-4daffba5a691')/ThingTypes"
            }
          },
          "MeasureMappings": {
            "results": [
              {
                "MappingId": "b807fece-b5b9-44ed-
a7b4-4daffba5a691",
                "SensorTypeName": "DemoReedSensorType",
                "CapabilityName": "DemoDoorOpenStatusCapability",

```

```

        "SensorTypeId":
"0b0aa063-3ddc-4920-8a8a-50744b4676d5",
        "CapabilityId": "e9bf8061-05fd-4d75-a002-
d6b2d9f45ef8",
        "ThingTypeName":
"iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType",
        "CapabilityPropertyId": "DemoDoorOpenStatus",
        "PropertySetTypeId":
"iotae.sandboxhdibd1.conv.services.demo.package:DoorOpenStatusPST",
        "NamedPropertySetTypeId": "DoorOpenStatusNPST",
        "NamedPropertySetTypePropertyId":
"DoorOpenStatus"
    },
    {
        "MappingId": "b807fece-b5b9-44ed-
a7b4-4daffba5a691",
        "SensorTypeName": "DemoCoolerSensorType",
        "CapabilityName": "DemoTemperatureCapability",
        "SensorTypeId":
"3376ab0a-34b2-4c46-877b-3d76f676d77f",
        "CapabilityId": "ee50a213-0878-4c88-
a6df-09fa00421c68",
        "ThingTypeName":
"iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType",
        "CapabilityPropertyId": "DemoHumidity",
        "PropertySetTypeId":
"iotae.sandboxhdibd1.conv.services.demo.package:TemperaturePST",
        "NamedPropertySetTypeId": "TemperatureNPST",
        "NamedPropertySetTypePropertyId": "HumidityProp"
    }
]
},
"TargetMappings": {
    "results": [
        {
            "MappingId": "b807fece-b5b9-44ed-
a7b4-4daffba5a691",
            "SensorTypeName": "DemoPressureSensorType",
            "CapabilityName": "DemoPressureCapability",
            "SensorTypeId": "5e70f20b-c0cb-4146-8c8b-
eeae96a86f01",
            "CapabilityId": "247adaca-2fbc-4003-
b2a6-96eaf0525962",
            "ThingTypeName":
"iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType",
            "CapabilityPropertyId": "DemoPressure",
            "PropertySetTypeId":
"iotae.sandboxhdibd1.conv.services.demo.package:ReferencePST",
            "NamedPropertySetTypeId": "CompressorNPST",
            "NamedPropertySetTypeReferencePropertyId":
"PressureTargetValue"
        },
        {
            "MappingId": "b807fece-b5b9-44ed-
a7b4-4daffba5a691",
            "SensorTypeName": "DemoPressureSensorType",
            "CapabilityName": "DemoPressureCapability",
            "SensorTypeId": "5e70f20b-c0cb-4146-8c8b-
eeae96a86f01",
            "CapabilityId": "247adaca-2fbc-4003-
b2a6-96eaf0525962",
            "ThingTypeName":
"iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType",
            "CapabilityPropertyId": "DemoRotSpeed",
            "PropertySetTypeId":
"iotae.sandboxhdibd1.conv.services.demo.package:ReferencePST",
            "NamedPropertySetTypeId": "CompressorNPST",

```

```

        "NamedPropertySetTypeReferencePropertyId":
        "rotSpeedTargetValue"
    }
    ]
    }
    ],
    "SensorTypes": {
        "__deferred": {
            "uri": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v2/
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package%3ADemoCoolerType') /
SensorTypes"
        }
    }
}
}
}

```

## Response

### Response Status and Error Codes

Code	Description
200	Details of a specific thing type with its sensor type mappings retrieved successfully

## 8.4.3 Read a Thing Type with its Sensor Types used in Mappings

With this method, you get the thing type and sensor types used in mappings.

### Request

**URI:** ThingConfiguration/v2/ThingTypes('<thing type name>')?\$expand=SensorTypes/SensorTypeCapabilities/Capabilities/Properties

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <thingconf>.r



## Request Headers

Header	Required	Values
Accept-Language	No	Language of the thing type description. The default language is <a href="#">en</a> .

## Request Parameters

Parameter	Required	Data Type	Description
Name	Yes	string	Name of the thing type

## Request Example

```
ThingConfiguration/v2/  
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType')?  
$expand=SensorTypes/SensorTypeCapabilities/Capabilities/Properties
```

## Payload

Format: *JSON*

```
{  
  "d": {  
    "__metadata": {  
      "id": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:443/  
ThingConfiguration/v2/ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package  
%3ADemoCoolerType')",  
      "uri": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:443/  
ThingConfiguration/v2/ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package  
%3ADemoCoolerType')",  
      "type": "com.sap.apptot.ThingType"  
    },  
    "Name": "iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType",  
    "PackageName": "iotae.sandboxhdibd1.conv.services.demo.package",  
    "Description": "DemoCooler",  
    "ExtendedThingType": null,  
    "ThingTypeCategory": null,  
    "PropertySets": {  
      "__deferred": {  
        "uri": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:  
443/ThingConfiguration/v2/  
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package%3ADemoCoolerType') /  
PropertySets"  
      }  
    },  
    "Descriptions": {  
      "__deferred": {  
        "uri": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:  
443/ThingConfiguration/v2/  
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package%3ADemoCoolerType') /  
Descriptions"  
      }  
    },  
    "ThingTemplates": {  
      "__deferred": {  
        "uri": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:  
443/ThingConfiguration/v2/  
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package%3ADemoCoolerType') /  
ThingTemplates"  
      }  
    }  
  }  
}
```

```

    },
    "SensorTypeMappings": {
      "__deferred": {
        "uri": "https://config-thing-sap.cfapps.sap.hana.ondemand.com:
443/ThingConfiguration/v2/
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package%3ADemoCoolerType') /
SensorTypeMappings"
      }
    },
    "SensorTypes": {
      "results": [
        {
          "__metadata": {
            "id": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypes('0b0aa063-3ddc-4920-8a8a-50744b4676d5')",
            "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypes('0b0aa063-3ddc-4920-8a8a-50744b4676d5')",
            "type": "com.sap.apptot.SensorType"
          },
          "id": "0b0aa063-3ddc-4920-8a8a-50744b4676d5",
          "Name": "DemoReedSensorType",
          "SensorTypeCapabilities": {
            "results": [
              {
                "__metadata": {
                  "id": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypeCapabilities(SensorTypeId='0b0aa063-3ddc-4920-8a8a-50744b4676d5',Capabi
lityId='e9bf8061-05fd-4d75-a002-d6b2d9f45ef8')",
                  "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypeCapabilities(SensorTypeId='0b0aa063-3ddc-4920-8a8a-50744b4676d5',Capabi
lityId='e9bf8061-05fd-4d75-a002-d6b2d9f45ef8')",
                  "type": "com.sap.apptot.SensorTypeCapability"
                },
                "SensorTypeId":
"0b0aa063-3ddc-4920-8a8a-50744b4676d5",
                "CapabilityId": "e9bf8061-05fd-4d75-a002-
d6b2d9f45ef8",
                "Type": "measure",
                "Capabilities": {
                  "__metadata": {
                    "id": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
Capabilities('e9bf8061-05fd-4d75-a002-d6b2d9f45ef8')",
                    "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
Capabilities('e9bf8061-05fd-4d75-a002-d6b2d9f45ef8')",
                    "type": "com.sap.apptot.Capability"
                  },
                  "id": "e9bf8061-05fd-4d75-a002-d6b2d9f45ef8",
                  "Name": "DemoDoorOpenStatusCapability",
                  "Properties": {
                    "results": [
                      {
                        "Name": "DemoDoorOpenStatus",
                        "CapabilityId":
"e9bf8061-05fd-4d75-a002-d6b2d9f45ef8",
                        "UnitOfMeasure": "ff"
                      }
                    ]
                  }
                }
              }
            ]
          }
        }
      ]
    }
  }
}

```

```

    },
    {
      "__metadata": {
        "id": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypes('3376ab0a-34b2-4c46-877b-3d76f676d77f')",
        "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypes('3376ab0a-34b2-4c46-877b-3d76f676d77f')",
        "type": "com.sap.apptot.SensorType"
      },
      "id": "3376ab0a-34b2-4c46-877b-3d76f676d77f",
      "Name": "DemoCoolerSensorType",
      "SensorTypeCapabilities": {
        "results": [
          {
            "__metadata": {
              "id": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypeCapabilities(SensorTypeId='3376ab0a-34b2-4c46-877b-3d76f676d77f',Capabi
lityId='ee50a213-0878-4c88-a6df-09fa00421c68')",
              "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypeCapabilities(SensorTypeId='3376ab0a-34b2-4c46-877b-3d76f676d77f',Capabi
lityId='ee50a213-0878-4c88-a6df-09fa00421c68')",
              "type": "com.sap.apptot.SensorTypeCapability"
            },
            "SensorTypeId":
"3376ab0a-34b2-4c46-877b-3d76f676d77f",
            "CapabilityId": "ee50a213-0878-4c88-
a6df-09fa00421c68",
            "Type": "measure",
            "Capabilities": {
              "__metadata": {
                "id": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
Capabilities('ee50a213-0878-4c88-a6df-09fa00421c68')",
                "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
Capabilities('ee50a213-0878-4c88-a6df-09fa00421c68')",
                "type": "com.sap.apptot.Capability"
              },
              "id": "ee50a213-0878-4c88-a6df-09fa00421c68",
              "Name": "DemoTemperatureCapability",
              "Properties": {
                "results": [
                  {
                    "Name": "DemoHumidity",
                    "CapabilityId":
"ee50a213-0878-4c88-a6df-09fa00421c68",
                    "UnitOfMeasure": "ff"
                  }
                ]
              }
            }
          }
        ]
      }
    },
    {
      "__metadata": {
        "id": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/SensorTypes('5e70f20b-
c0cb-4146-8c8b-eeae96a86f01')",
        "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/SensorTypes('5e70f20b-
c0cb-4146-8c8b-eeae96a86f01')",

```

```

        "type": "com.sap.apptot.SensorType"
    },
    "id": "5e70f20b-c0cb-4146-8c8b-eeae96a86f01",
    "Name": "DemoPressureSensorType",
    "SensorTypeCapabilities": {
        "results": [
            {
                "__metadata": {
                    "id": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypeCapabilities(SensorTypeId='5e70f20b-c0cb-4146-8c8b-
eeae96a86f01',CapabilityId='247adaca-2fbc-4003-b2a6-96eaf0525962')",
                    "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
SensorTypeCapabilities(SensorTypeId='5e70f20b-c0cb-4146-8c8b-
eeae96a86f01',CapabilityId='247adaca-2fbc-4003-b2a6-96eaf0525962')",
                    "type": "com.sap.apptot.SensorTypeCapability"
                },
                "SensorTypeId": "5e70f20b-c0cb-4146-8c8b-
eeae96a86f01",
                "CapabilityId": "247adaca-2fbc-4003-
b2a6-96eaf0525962",
                "Type": "command",
                "Capabilities": {
                    "__metadata": {
                        "id": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
Capabilities('247adaca-2fbc-4003-b2a6-96eaf0525962')",
                        "uri": "https://config-thing-
sap.cfapps.sap.hana.ondemand.com:443/ThingConfiguration/v2/
Capabilities('247adaca-2fbc-4003-b2a6-96eaf0525962')",
                        "type": "com.sap.apptot.Capability"
                    },
                    "id": "247adaca-2fbc-4003-b2a6-96eaf0525962",
                    "Name": "DemoPressureCapability",
                    "Properties": {
                        "results": [
                            {
                                "Name": "DemoPressure",
                                "CapabilityId":
"247adaca-2fbc-4003-b2a6-96eaf0525962",
                                "UnitOfMeasure": "ff"
                            },
                            {
                                "Name": "DemoRotSpeed",
                                "CapabilityId":
"247adaca-2fbc-4003-b2a6-96eaf0525962",
                                "UnitOfMeasure": "ff"
                            }
                        ]
                    }
                }
            }
        ]
    }
}

```

## Response

### Response Status and Error Codes

Code	Description
200	Details of a specific thing type with its sensor types used in mappings, retrieved successfully

## 8.4.4 Update Sensor Type Mappings or Sensor Types in the Thing Type

With this method, you can update the sensor type mappings in the thing type.

### Sensor Type Mappings

The valid operations are `ADD`, `DELETE`, and `UPDATE`.

The operations are available for overall sensor type mapping, and also to measure mappings and target mappings in the mappings.

### Overall Sensor Type Mappings

The valid operations are `ADD`, `DELETE`, and `UPDATE`.

`ADD`: Adds a new sensor type mapping to the existing list of sensor type mappings

`UPDATE`: Updates the name or descriptions of an existing sensor type mapping

`DELETE`: Deletes the mapping from the existing list of mappings

### Measure Mapping Inside a Sensor Type Mapping

The valid operations are `ADD`, `DELETE`, and `UPDATE`.

`ADD`: Adds a new measure mapping to the existing list of measure mappings

`UPDATE`: Updates the existing measure mapping

`DELETE`: Deletes the existing measure mapping

## Target Mapping Inside a Sensor Type Mapping

The valid operations are `ADD`, `DELETE`, and `UPDATE`.

`ADD`: Adds a new target mapping to the existing list of target mappings

`UPDATE`: Updates the existing target mapping

`DELETE`: Deletes the existing target mapping

## Sensor Types

The valid operations are `ADD`, `DELETE`, and `UPDATE`.

The operations are available for overall sensor types, and also capability assignment within the sensor type.

## Overall Sensor Type

`ADD`: Creates a sensor type in IoT services

`UPDATE`: Updates the sensor type name in IoT services

`DELETE`: Deletes the sensor type from IoT services

## Capability Assignment within the Sensor Type

`ADD`: Creates a new capability, and assigns to sensor type in IoT services

`UPDATE`: Updates the capability type to **measure** or **command**

`DELETE`: Unassigns the capability from the sensor type

## Request

**URI:** `/ThingConfiguration/v2/ThingTypes(<Thing Type Name>)`

**Operation Type:** `CRUD`

**HTTP Method:** `PUT`

**Permissions:** `<thingconf>.u`

## Request Headers

Header	Required	Values
Accept-Language	No	Language of the thing type description. The default language is <a href="#">en</a> .
If-Match	Yes	Latest ETag value of a specific thing type that is updated using the PATCH request.

## Request Parameters

Parameter	Required	Data Type	Description
thing type name	Yes	String	Name of the thing type

## Request Example

In sensor types, the below request creates a new capability `DemoTemperatureCapability`, and creates a new sensor type `DemoThermocoupleSensorType`, and assigns the capability `DemoTemperatureCapability` to it.

In sensor type mappings, the below request deletes a mapping from the target mapping, and adds a new mapping with the sensor type `DemoThermocoupleSensorType`.

```
/ThingConfiguration/v2/  
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType')
```

## Payload

Format: [JSON](#)

```
{  
  "Name": "iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType",  
  "Descriptions": [{  
    "LanguageCode": "en",  
    "Description": "ConvCooler"  
  }],  
  "SensorTypeMappings": [{  
    "Descriptions": [{  
      "LanguageCode": "en",  
      "Description": "Sensor Type Mappings"  
    }],  
    "MappingId": "b807fece-b5b9-44ed-a7b4-4daffba5a691",  
    "Name": "DemoConvCoolerSensorTypeMappings",  
    "MeasureMappings": [  
      {  
        "MappingId": "b807fece-b5b9-44ed-a7b4-4daffba5a691",  
        "SensorTypeName": "DemoThermocoupleSensorType",  
        "CapabilityName": "DemoTemperatureCapability",  
        "CapabilityPropertyId": "DemoTemperature",  
        "PropertySetTypeId":  
        "iotae.sandboxhdibd1.conv.services.demo.package:TemperaturePST",  
        "NamedPropertySetTypeId": "TemperatureNPST",  
        "NamedPropertySetTypePropertyId": "TemperatureProp",  
        "Operation": "ADD"  
      }  
    ],  
    "TargetMappings":
```

```
[
  {
    "MappingId": "b807fece-b5b9-44ed-
a7b4-4daffba5a691",
    "SensorTypeName": "DemoPressureSensorType",
    "CapabilityName": "DemoPressureCapability",
    "SensorTypeId": "5e70f20b-c0cb-4146-8c8b-
eeae96a86f01",
    "CapabilityId": "247adaca-2fbc-4003-
b2a6-96eaf0525962",
    "ThingTypeName":
"iotae.sandboxhdbd1.conv.services.demo.package:DemoCoolerType",
    "CapabilityPropertyId": "DemoRotSpeed",
    "PropertySetTypeId":
"iotae.sandboxhdbd1.conv.services.demo.package:ReferencePST",
    "NamedPropertySetTypeId": "CompressorNPST",
    "NamedPropertySetTypeReferencePropertyId":
"rotSpeedTargetValue",
    "Operation": "DELETE"
  }
],
{
  "SensorTypes": [{
    "Name": "DemoThermocoupleSensorType",
    "Operation": "ADD",
    "SensorTypeCapabilities": [{
      "Capability": {
        "Name": "DemoTemperatureCapability",
        "Properties": [{
          "Name": "DemoTemperature",
          "DataType": "Double",
          "UnitOfMeasure": "C"
        }]
      },
      "Type": "measure",
      "Operation": "ADD"
    }]
  }]
}
```

## Response

### Response Status and Error Codes

Code	Description
204	The sensor type mappings in the thing type updated successfully

## 8.4.5 Delete a Thing Type with its Sensor Type Mappings

With this method, you can delete the thing type and all the mappings associated to it.

The **ETag** value is mandatory to delete a thing type. You must retrieve the **ETag** value of the thing type using the GET (`/ThingConfiguration/v2/ThingTypes('<thing type name>')`) service. The **ETag** value is assigned to the request header parameter **If-Match** in the **PATCH** request. If you do not include the **ETag**



value in the request header, the server sends HTTP response code 428. If the `ETag` value is not correct, the server sends HTTP response code 412.

#### Note

- You cannot delete a thing type if one of the following reasons is true:
  - Extended to another thing type
  - Things are already created
- If you delete a thing type which has been mapped to one or more sensor types, the system automatically deletes all the mappings referring to that thing type.

## Request

**URI:** `/ThingConfiguration/v2/ThingTypes(<Thing Type Name>)`

**Operation Type:** CRUD

**HTTP Method:** `DELETE`

**Permissions:** `<thingconf>.d`

### Request Headers

Header	Required	Values
If-Match	Yes	Latest ETag value of a specific thing type that is updated using the PATCH request.

### Request Parameters

Parameter	Required	Data Type	Description
thing type name	Yes	String	Unique name of the thing type

### Request Example

```
/ThingConfiguration/v2/  
ThingTypes('iotae.sandboxhdibd1.conv.services.demo.package:DemoCoolerType')
```

## Response

### Response Status and Error Codes

Code	Description
204	Thing Type and its mappings deleted successfully

## 9 Custom Master Data

In Intelligent Enterprises, business benefit is derived by combining sensor data and business data. This is achieved by enriching sensor data by adding business context to it. Business context can come in different forms. It can be master data information (such as Material, Customer, Vendor, Employee, Business Partner, Product, and so on) or it can be information about business transactions (Sales Order, Goods Receipt, Service Request, Purchase Requisition, and so on). Supporting master data and business transactions is essential for customers to realize the true business value.

Things are often related to backend master data objects and business transaction objects as part of the E2E process in the Intelligent Enterprise. To enable generic things to be integrated into business processes, associations between thing instances and master data objects are requested.

### Pagination

Retrieving more number of objects than the recommended limits with only one collective GET request can lead to a high system load and a significant increase of response times. To avoid performance decline, we limit the number of objects to be retrieved in a single collective GET call, using the concept of paging. The default value of server size paging is set to 500 from next release, and this can still be modified using the paging options such as `$top` and `$skip`, but however the value will not exceed the server size page value.

`$top`: With the query option `$top`, the user can specify the maximum number of entries that should be returned, starting from the beginning. The default value of `$top`, if not explicitly mentioned, will be equal to the size of the server paging (in this case 500).

`$skip`: With the query option `$skip`, the user can specify the number of entries that should be ignored at the beginning of a collection. So if a user specifies `$skip=n`, then this returns the list of entries starting at position `n+1`. The default value of skip would be skip token, if passed, else the value of the server size paging is considered.

### 9.1 Object Types

Define a master data or transaction data object type.

A master data object type is defined with atleast:

- the object type
- the language dependent object type description

In addition to the above, you can define upto 5 property types per object type, which support the standard data types (not large string, for example) as well as the enumerations with language dependent descriptions.

### Note

Deletion of a master data object type shall only be allowed if there is no instance existing.

**Resource Path:** `http://<server address>[:<port number>][/path]/CustomDataConfiguration`

**Example for a base URI in a cloud foundry environment:** `https://sap-iotaexplore.iot-sap.cfapps.eu10.hana.ondemand.com/config-customdata-sap/CustomDataConfiguration`

## Operations

### CRUD Operations

HTTP Method	Operation	URI
GET	<a href="#">Read an Object Type [page 606]</a>	<code>/CustomDataConfiguration/v1/ObjectTypes('&lt;ObjectTypeName&gt;')</code>
GET	<a href="#">Read all Object Types of a Package [page 615]</a>	<code>/CustomDataConfiguration/v1/Packages('&lt;PackageID&gt;')/ObjectTypes</code>
GET	<a href="#">Read all Object Types [page 611]</a>	<code>/CustomDataConfiguration/v1/ObjectTypes</code>
POST	<a href="#">Create an Object Type [page 599]</a>	<code>/CustomDataConfiguration/v1/Packages('&lt;PackageID&gt;')/ObjectTypes</code>
POST	<a href="#">Update an Object Type [page 604]</a>	<code>/CustomDataConfiguration/v1/ObjectTypes('&lt;ObjectTypeName&gt;')</code>
DELETE	<a href="#">Delete an Object Type [page 620]</a>	<code>/CustomDataConfiguration/v1/ObjectTypes('&lt;ObjectTypeName&gt;')</code>

### 9.1.1 Create an Object Type

This operation helps to create a new object type with a maximum of five properties, along with translations in different languages. For example, Material is an object type having Density and Quantity as its properties.

### Note

- In the configuration, you can define any number of object types. However, you can create only one object type per POST request. You can create more number of object types for a package through multiple POST requests.

### Naming convention for object types

- It is always prefixed with the package name.
- It can be a combination of alphanumeric characters and underscore.
- Each name must be fully qualified and contains a package name, colon as a separator and the object name. Maximum length of fully qualified name is 81 characters.

- Do not start the object name with a digit or an underscore.
- Length should be at least 3 characters.
- Camel case is generally allowed, however there is a uniqueness check carried out against textual case conflicts:
  - Two names with same characters of different textual case is not allowed. For example: A value "AbcDef" is not accepted when there is another value "abcDEF" that exists.
- Verify that a colon is used only to separate a package name and the object type name. The syntax is: <package>:<name>. For example: core.rawmaterial:material
- Name must be unique within a package.
- Only the following characters are supported:
  - Uppercase and lowercase alphabets *a* through *z* and *A* through *Z*
  - Numeric digits 0 through 9
  - Punctuation marks underscore (\_), hyphen (-), colon (:), and full-stop (.)
- Maximum length is 50 characters.
- Property names have to be unique for a given set of object types.

## Annotation

Annotation is an array of attributes that are used to qualify the properties of an Object Type. Annotations can be defined both for an Object Type and for its Property type.

## Annotation for an Object Type

For an Object Type, annotation can be used to mark the Object Type as sensitive or personal.

Only one of the following annotations is supported for an Object Type:

- `com.sap.apptot.security:pii` (personal information)  
You can assign this annotation to an Object Type that contains data to identify a person or subject. Data such as nick name, birth date, and email address are few examples that can be categorized as data that identifies a person.
- `com.sap.apptot.security:spi` (sensitive personal information)  
You can assign this annotation to an Object Type that contains critical data of a person. Data such as identification number, ethnic origin, and biometric data are few examples that can be categorized as critical data of a person.  
After you define the annotations, all properties of the object type are considered as personal data or sensitive data based on the annotations defined. You can remove or add the annotation for the object type, only if data does not exist for the properties of the object type. Whenever you create or modify data for properties that are identified as personal data or sensitive data, the system will generate audit logs.

Example to create Object Type with Annotation

**API:** /CustomDataConfiguration/v1/Packages('iotae.devtest1.cdemo')/ObjectTypes

```
{
  "Name": "iotae.devtest1.cdemo:PartnerInfo",
  "PackageName": "iotae.devtest1.cdemo",
  "Descriptions": [
    {
      "LanguageCode": "en",
      "Description": "Person Data"
    }
  ],
  "Source": "ERP",
  "IsMultiLingual": false,
```

```

    "Annotations": [
      {
        "Name": "com.sap.appiot.security:pii"
      }
    ],
    "Properties": [
      {
        "Name": "Partner_Name",
        "Type": "String",
        "PropertyLength": "130"
      }
    ]
  }
}

```

## Annotation for a Property of Object Type

You can define annotation in a package (refer the thing configuration API [Annotation \[page 501\]](#)) and assign it to the property of an object type that belongs to the same package of an Annotation.

Example to create an Object Type with Property Annotation (assuming `iotae.devtest1.cdemo:CustomAnnotation1` is already created using thing configuration API)

**API:** `/CustomDataConfiguration/v1/Packages('iotae.devtest1.cdemo')/ObjectTypes`

```

{
  "Name": "iotae.devtest1.cdemo:PartnerInfo",
  "PackageName": "iotae.devtest1.cdemo",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Person Data"
  }],
  "Source": "ERP",
  "IsMultiLingual": false,
  "Properties": [{
    "Name": "Partner_Name",
    "Type": "String",
    "PropertyLength": "130",
    "Annotations": [{
      "Name": "iotae.devtest1.cdemo:CustomAnnotation1"
    }]
  }]
}

```

## Request

**URI:** `/CustomDataConfiguration/v1/Packages('<package name>')/ObjectTypes`

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** `<cdconf>.c`

## Request Headers

Header	Required	Values
Accept-Language	No	Language of the object type description. The default language is <a href="#">en</a> .

## Request Parameters

Parameter	Required	Data Type	Description
package name	Yes	String	Unique name of the package

## Request Example

/CustomDataConfiguration/v1/Packages('iotae.devtest1.cdemo')/ObjectTypes

## Payload

**Format:** JSON

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following example.

### Example

/CustomDataConfiguration/v1/Packages('iotae.devtest1.cdemo')/ObjectTypes

```
{
  "Name": "iotae.devtest1.cdemo:Material",
  "PackageName": "iotae.devtest1.cdemo",
  "Descriptions": [
    {
      "LanguageCode": "en",
      "Description": "Material Data"
    }
  ],
  "Source": "Material Management",
  "IsMultiLingual": true,
  "Properties": [
    {
      "Name": "MATNR",
      "Type": "String",
      "PropertyLength": "120",
      "IsMultiLingual": false
    }, {
      "Name": "Material_Type",
      "Type": "String",
      "PropertyLength": "120",
      "IsMultiLingual": true
    }
  ]
}
```

## Response

### Response Headers

Header	Description
ETag	87FF06A3B6784A3B9F4A6EC3F9EA0675

## Response Status and Error Codes

Code	Description
201	Object type created successfully for a specific package

## Response Payload Example

```
{
  "d": {
    "__metadata": {
      "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
      "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
      "type": "com.sap.apptot.ObjectType"
    },
    "Name": "iotae.devtest1.cdemo:Material",
    "PackageName": "iotae.devtest1.cdemo",
    "Description": "",
    "IsMultiLingual": true,
    "NumberOfProperties": null,
    "Source": "Material Management",
    "Annotations": {
      "__deferred": {
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Annotations"
      }
    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Descriptions"
      }
    },
    "Properties": {
      "__deferred": {
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Properties"
      }
    },
    "ObjectTypeAssociations": {
      "__deferred": {
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/ObjectTypeAssociations"
      }
    }
  }
}
```

With Annotation

### Example

/CustomDataConfiguration/v1/Packages('iotae.devtest1.cdemo')/ObjectTypes

```
{
  "Name": "iotae.devtest1.cdemo:Material1",
  "PackageName": "iotae.devtest1.cdemo",
  "Descriptions": [
```

```

    {
      "LanguageCode": "en",
      "Description": "Material Data"
    }
  ],
  "Source": "Material Management",
  "IsMultiLingual": true,
  "Annotations": [
    {
      "Name": "com.sap.appiot.security:pii"
    }
  ],
  "Properties": [
    {
      "Name": "MATNR",
      "Type": "String",
      "PropertyLength": "120",
      "IsMultiLingual": false
    }, {
      "Name": "Material_Type",
      "Type": "String",
      "PropertyLength": "120",
      "IsMultiLingual": true
    }
  ]
}

```

## 9.1.2 Update an Object Type

With this method, you can update the details of the specified object type.

You can update the object type as follows:

- You can add annotation, delete or change annotation that is already assigned to the object type.
- You can add new properties, edit the existing properties, or delete the existing properties.
- You can add, change, or delete the annotation assigned to the property of an object type.
- You can add the field `IsMultiLingual=true` to support multiple languages for its string type properties, if the object type was not already supporting multiple languages. This is possible only if there is no data already existing for those properties.
- You can only increase the length of the property, if data already exists for the property and the data type is **String**. For any other data types, you cannot change the property.
- You can add new properties by reusing the property types, if the following conditions are true:
  - Property type and object type belongs to the same package.

The `Etag` value is mandatory to update an object type. You must retrieve the `Etag` value of the object type using the `GET (/CustomDataConfiguration/v1/ObjectTypes('<object type name>'))` service. The `Etag` value is assigned to the request header parameter `If-Match` in the `PATCH` request. If you do not include the `Etag` value in the request header, the server sends HTTP response code 428. If the `Etag` value is not correct, the server sends HTTP response code 412.

After updating an object type, the system generates a new `Etag` value for the object type and replaces the old value with the new value.

### Note

- You cannot change the name of the object type.



- If you are not editing the already assigned annotation for a property, you must still specify the annotation details for the property in the request payload. If you do not specify the details, the service deletes the assignment and the annotation is no more assigned to that property.
- You cannot change the value of the `IsMultiLingual` field for an object type, if data already exists for its properties.
- The valid operations for the properties are ADD, DELETE, and UPDATE.
  - You cannot change the value of the field `IsMultiLingual`, if data already exists for that property.
  - You can add the field `IsMultiLingual=true` or `IsMultiLingual=false` for the string type property, if data does not exist and the object type supports multiple languages.
- The valid operations for the annotations of the object type are ADD, DELETE, and UPDATE.
  - You can add annotation for an object type if nothing is already assigned and even though the data exists for the properties.
  - You cannot add multiple annotations for the object type.
  - You cannot delete or change the already assigned annotation, if data exists for the properties of the object type.

## Request

**URI:** `/CustomDataConfiguration/v1/ObjectTypes('<ObjectName>')`

**Operation Type:** CRUD

**HTTP Method:** [PATCH](#)

**Permissions:** `<cdconf>.u`

## Request Headers

Header	Required	Values
<code>Accept-Language</code>	No	Language of the object type description. The default language is <a href="#">en</a> .
<code>If-Match</code>	Yes	Latest Etag value of a specific object type name that is updated using the <a href="#">PATCH</a> request.

## Request Parameters

Parameter	Required	Data Type	Description
<code>object type name</code>	Yes	String	Name of the object type

## Request Example

`/CustomDataConfiguration/v1/ObjectTypes('iotae.devtest1.demo:Material')`

## Payload

**Format:** JSON

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "Name": "iotae.devtest1.cdemo:Material1",
  "PackageName": "iotae.devtest1.cdemo",
  "Descriptions": [
    {
      "LanguageCode": "en",
      "Description": "Material Data"
    }
  ],
  "Source": "Material Management",
  "IsMultiLingual": true,
  "Properties": [
    {
      "Operation": "ADD",
      "Name": "Material_Description",
      "Type": "String",
      "PropertyLength": "120",
      "IsMultiLingual": true
    },
    {
      "Operation": "DELETE",
      "Name": "Material_Type"
    }
  ]
}
```

### Note

You cannot update the object type name using the `PATCH` request. The object type name specified in the request payload is ignored for update.

## Response

### Response Status and Error Codes

Code	Description
204	Details of the specified object type updated successfully

## 9.1.3 Read an Object Type

With this method, you can retrieve the details of a specific object type.

In order to read a given object type, we have to provide the object type name.

## Request

**URI:** /CustomDataConfiguration/v1/ObjectTypes('<ObjectName>')

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <cdconf>.r

### Request Parameters

Parameter	Required	Data Type	Description
\$expand	No	String	Helps to get all the properties for the specified object type

### Request Example

```
/CustomDataConfiguration/v1/ObjectTypes('iotae.devtest1.cdemo:Material')
```

## Response

### Response Headers

Header	Value
Etag	DC68FB05A8AB46619F7F6E22F5F5241F

### Response Status and Error Codes

Code	Description
200	Details of a specific object type retrieved successfully

### Response Payload Example

```
/CustomDataConfiguration/v1/  
ObjectTypes('iotae.devtest1.aejob.automobilestes:testOT1308')/Properties
```

```
{  
  "d": {  
    "results": [  
      {  
        "_metadata": {  
          "id": "https://config-customdata-sap-  
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/  
Properties(Name='testProperty1',ObjectType='iotae.devtest1.aejob.automobilestes  
%3AtestOT1308')",  
          "uri": "https://config-customdata-sap-  
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/  
Properties(Name='testProperty1',ObjectType='iotae.devtest1.aejob.automobilestes  
%3AtestOT1308')",  
          "type": "com.sap.appiot.Property"  
        },  
        "Name": "testProperty1",  
      }  
    ]  
  }  
}
```

```

        "ObjectType": "iotae.devtest1.aejob.automobilestes:testOT1308",
        "Description": "Test Object Type 7",
        "Type": "String",
        "PropertyLength": "128"
    },
    {
        "__metadata": {
            "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
Properties(Name='testProperty2',ObjectType='iotae.devtest1.aejob.automobilestes
%3AtestOT1308')",
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
Properties(Name='testProperty2',ObjectType='iotae.devtest1.aejob.automobilestes
%3AtestOT1308')",
            "type": "com.sap.apptot.Property"
        },
        "Name": "testProperty2",
        "ObjectType": "iotae.devtest1.aejob.automobilestes:testOT1308",
        "Description": "Test Object Type 8",
        "Type": "String",
        "PropertyLength": "128"
    }
]
}
}

```

/CustomDataConfiguration/v1/ObjectTypes('iotae.devtest1.cdemo:Material')

```

{
    "d": {
        "__metadata": {
            "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
            "type": "com.sap.apptot.ObjectType"
        },
        "Name": "iotae.devtest1.cdemo:Material",
        "PackageName": "iotae.devtest1.cdemo",
        "Description": "Material Data",
        "IsMultiLingual": true,
        "NumberOfProperties": null,
        "Source": "Material Management",
        "Annotations": {
            "__deferred": {
                "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Annotations"
            }
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Descriptions"
            }
        },
        "Properties": {
            "__deferred": {
                "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Properties"
            }
        }
    }
}

```

```

      "ObjectTypeAssociations": {
        "__deferred": {
          "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/ObjectTypeAssociations"
        }
      }
    }
  }
}

```

With \$expand=Properties

```

/CustomDataConfiguration/v1/ObjectTypes('iotae.devtest1.cdemo:Material')?
$format=json&$expand=Properties

```

```

{
  "d": {
    "__metadata": {
      "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
      "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
      "type": "com.sap.apptot.ObjectType"
    },
    "Name": "iotae.devtest1.cdemo:Material",
    "PackageName": "iotae.devtest1.cdemo",
    "Description": "Material Data",
    "IsMultiLingual": true,
    "NumberOfProperties": null,
    "Source": "Material Management",
    "Annotations": {
      "__deferred": {
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Annotations"
      }
    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Descriptions"
      }
    },
    "Properties": {
      "results": [
        {
          "Name": "MATNR",
          "ObjectType": "iotae.devtest1.cdemo:Material",
          "Description": "",
          "Type": "String",
          "PropertyLength": "120",
          "UnitOfMeasure": "",
          "IsMultiLingual": false,
          "PropertyType": ""
        },
        {
          "Name": "Material_Type",
          "ObjectType": "iotae.devtest1.cdemo:Material",
          "Description": "",
          "Type": "String",
          "PropertyLength": "120",
          "UnitOfMeasure": "",
          "IsMultiLingual": true,
          "PropertyType": ""
        }
      ]
    }
  }
}

```

```

    }
  ],
  "ObjectTypeAssociations": {
    "__deferred": {
      "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/ObjectTypeAssociations"
    }
  }
}

```

With \$expand=Annotations

```

/CustomDataConfiguration/v1/ObjectTypes('iotae.devtest1.cdemo:Material')?
$format=json&$expand=Annotations

```

```

{
  "d": {
    "__metadata": {
      "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
      "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
      "type": "com.sap.apptot.ObjectType"
    },
    "Name": "iotae.devtest1.cdemo:Material",
    "PackageName": "iotae.devtest1.cdemo",
    "Description": "Material Data",
    "IsMultiLingual": true,
    "NumberOfProperties": null,
    "Source": "Material Management",
    "Annotations": {
      "results": [
        {
          "Name": "com.sap.apptot.security:pii",
          "PackageName": "com.sap.apptot.security",
          "Description": null
        }
      ]
    },
    "Descriptions": {
      "__deferred": {
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Descriptions"
      }
    },
    "Properties": {
      "__deferred": {
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Properties"
      }
    },
    "ObjectTypeAssociations": {
      "__deferred": {
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/ObjectTypeAssociations"
      }
    }
  }
}

```

```
}
```

## 9.1.4 Read all Object Types

With this method, you can read all the object types.

### Request

**URI:** /CustomDataConfiguration/v1/ObjectTypes

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <cdconf>.r

### Request Headers

Header	Required	Values
Accept-Language	No	Language of the object type description. The default language is <i>en</i> .
sap-iot-propertiescount	No	Include a new field <code>NumberOfProperties</code> in the response payload  Only if you specify the value <i>true</i> for this parameter, the system includes the count of properties in a new field <code>NumberOfProperties</code> in the response payload. If you specify <i>false</i> or do not use this header parameter, the system does not return the count of properties in the response payload.

### Query Request Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set  If \$orderby is not chosen, the results will be sorted by ObjectType in the ascending order.
\$skip	No	Integer	Number of the first n records to be excluded from the result set  If \$orderby is not chosen, the results will be sorted by ObjectType in the ascending order.

Parameter	Required	Data Type	Description
\$inlinecount	No	String	<p>Total number of object types available; permissible values are <i>allpages</i> and <i>none</i>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"> <li>• If you use this with the \$top and \$skip parameters, the result contains the total count of the object types available.</li> <li>• If you use this with \$filter parameter, the result contains the total count of the object types based on the defined filter condition.</li> </ul>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order.</p> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; only valid fields are object type name, package name, and description. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the separators – <i>and</i>, <i>or</i>.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using \$filter. Valid fields are object type name, package name, and description.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are object type name, package name, and description.</p>

## Request Example

/CustomDataConfiguration/v1/ObjectTypes

## Response

### Response Status and Error Codes

Code	Description
200	Details of all object types retrieved successfully

### Response Payload Example

Format: *JSON*



/CustomDataConfiguration/v1/ObjectTypes

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
          "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
          "type": "com.sap.apptot.ObjectType"
        },
        "Name": "iotae.devtest1.cdemo:Material",
        "PackageName": "iotae.devtest1.cdemo",
        "Description": "Material Data",
        "IsMultiLingual": true,
        "Source": "Material Management",
        "Annotations": {
          "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Annotations"
          }
        },
        "Descriptions": {
          "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Descriptions"
          }
        },
        "Properties": {
          "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Properties"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')",
          "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')",
          "type": "com.sap.apptot.ObjectType"
        },
        "Name": "iotae.devtest1.cdemo:Material1",
        "PackageName": "iotae.devtest1.cdemo",
        "Description": "Material Data",
        "IsMultiLingual": true,
        "Source": "Material Management",
        "Annotations": {
          "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Annotations"
          }
        },
        "Descriptions": {
          "__deferred": {
```

```

        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Descriptions"
    },
    "Properties": {
        "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Properties"
        }
    }
}
]
}
}

```

The following is an example payload with the count of properties included in the response payload:

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
          "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
          "type": "com.sap.appiot.ObjectType"
        },
        "Name": "iotae.devtest1.cdemo:Material",
        "PackageName": "iotae.devtest1.cdemo",
        "Description": "Material Data",
        "Source": "Material Management",
        "NumberOfProperties": 2,
        "IsMultiLingual": true,
        "Annotations": {
          "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Annotations"
          }
        },
        "Descriptions": {
          "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Descriptions"
          }
        },
        "Properties": {
          "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Properties"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')",

```

```

        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')",
        "type": "com.sap.appiot.ObjectType"
    },
    "Name": "iotae.devtest1.cdemo:Material1",
    "PackageName": "iotae.devtest1.cdemo",
    "Description": "Material Data",
    "IsMultiLingual": true,
    "Source": "Material Management",
    "NumberOfProperties": 2,
    "Annotations": {
        "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Annotations"
        }
    },
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Descriptions"
        }
    },
    "Properties": {
        "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Properties"
        }
    }
}
]
}
}

```

## 9.1.5 Read all Object Types of a Package

With this method, you can read all object types of the specified package.

### Request

**URI:** /CustomDataConfiguration/v1/Packages('<package name>')/ObjectTypes

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <cdconf>.r

## Request Headers

Header	Required	Values
Accept-Language	No	Language of the property set type description. The default language is <a href="#">en</a> .
sap-iot-propertiescount	No	<p>Include a new field <code>NumberOfProperties</code> in the response payload.</p> <p>Only if you specify the value <a href="#">true</a> for this parameter, the system includes the count of properties in a new field <code>NumberOfProperties</code> in the response payload. If you specify <a href="#">false</a> or do not use this header parameter, the system does not return the count of properties in the response payload.</p>

## Request Parameters

Parameter	Required	Data Type	Description
package name	Yes	String	Unique name of the package

## Query Request Parameters

Parameter	Required	Type	Description
\$top	No	Integer	<p>Number of records to include in the result set</p> <p>If \$orderby is not chosen, the results will be sorted by ObjectType in the ascending order.</p>
\$skip	No	Integer	Number of the first n records to be excluded from the result set
\$inlinecount	No	String	<p>Total number of object types available; permissible values are <a href="#">allpages</a> and <a href="#">none</a>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"><li>• If you use this with the \$top and \$skip parameters, the result contains the total count of the object types available.</li><li>• If you use this with \$filter parameter, the result contains the total count of object types based on the defined filter condition.</li></ul>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order.</p> <p>To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</p>

Parameter	Required	Type	Description
\$filter	No	String	<p>Filter condition to be applied; the only valid fields are object type name and description. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the separators – <i>and</i>, <i>or</i>.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using \$filter. Valid fields are object type name and description.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are object type name, package name and description.</p>

## Request Example

```
/CustomDataConfiguration/v1/Packages('iotae.devtest1.aejob.automobilestes')/
ObjectTypes
```

## Response

### Response Status and Error Codes

Code	Description
200	Details of all object types retrieved successfully

### Response Payload Example

```
/CustomDataConfiguration/v1/Packages('iotae.devtest1.cdemo')/ObjectTypes
```

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
          "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
          "type": "com.sap.apptot.ObjectType"
        },
        "Name": "iotae.devtest1.cdemo:Material",
        "PackageName": "iotae.devtest1.cdemo",
        "Description": "Material Data",
        "IsMultiLingual": true,
        "Source": "Material Management",
        "Annotations": {
          "__deferred": {
```

```

        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Annotations"
    },
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Descriptions"
        }
    },
    "Properties": {
        "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Properties"
        }
    }
},
{
    "__metadata": {
        "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')",
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')",
        "type": "com.sap.apptot.ObjectType"
    },
    "Name": "iotae.devtest1.cdemo:Material1",
    "PackageName": "iotae.devtest1.cdemo",
    "Description": "Material Data",
    "IsMultiLingual": true,
    "Source": "Material Management",
    "Annotations": {
        "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Annotations"
        }
    },
    "Descriptions": {
        "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Descriptions"
        }
    },
    "Properties": {
        "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Properties"
        }
    }
}
]
}
}

```

The following is an example payload with the count of properties included in the response payload:

```

{
  "d": {
    "results": [
      {

```

```

        "__metadata": {
            "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')",
            "type": "com.sap.apptot.ObjectType"
        },
        "Name": "iotae.devtest1.cdemo:Material",
        "PackageName": "iotae.devtest1.cdemo",
        "Description": "Material Data",
        "Source": "Material Management",
        "NumberOfProperties": 2,
        "IsMultiLingual": true,
        "Annotations": {
            "__deferred": {
                "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Annotations"
            }
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Descriptions"
            }
        },
        "Properties": {
            "__deferred": {
                "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial')/Properties"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')",
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')",
            "type": "com.sap.apptot.ObjectType"
        },
        "Name": "iotae.devtest1.cdemo:Material1",
        "PackageName": "iotae.devtest1.cdemo",
        "Description": "Material Data",
        "IsMultiLingual": true,
        "Source": "Material Management",
        "NumberOfProperties": 2,
        "Annotations": {
            "__deferred": {
                "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Annotations"
            }
        },
        "Descriptions": {
            "__deferred": {
                "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Descriptions"
            }
        },
        "Properties": {

```

```

      "_deferred": {
        "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypes('iotae.devtest1.cdemo%3AMaterial1')/Properties"
      }
    }
  ]
}

```

## 9.1.6 Delete an Object Type

With this method, you can delete the specified object type.

The *Etag* value is mandatory to delete an object type. You must retrieve the *Etag* value of the object type using the `GET(/CustomDataConfiguration/v1/ObjectTypes('<object type name>'))` service. The *Etag* value is assigned to the request header parameter `If-Match` in the *DELETE* request. If you do not include the *Etag* value in the request header, the server sends HTTP response code 428. If the *Etag* value is not correct, the server sends HTTP response code 412.

To delete an object type, you have to provide the name of the object type to be deleted. For security reasons, the latest *Etag* has to be sent as an `If-Match` header. For example, `If-Match - 87FF06A3B6784A3B9F4A6EC3F9EA0675`

### Note

Once the object type is deleted successfully, you will get a 204 No Content Response.

## Request

**URI:** `/CustomDataConfiguration/v1/ObjectTypes('<ObjectTypeName>')`

**Operation Type:** CRUD

**HTTP Method:** *DELETE*

**Permissions:** `<cdconf>.d`

## Request Headers

Header	Required	Values
<code>If-Match</code>	Yes	Latest <i>Etag</i> value of a specific object type that is updated using the <i>PATCH</i> request.



## Request Parameters

Parameter	Required	Data Type	Description
object type name	Yes	String	Unique name of the object type

## Request Example

```
/CustomDataConfiguration/v1/ObjectTypes('iotae.devtest1.cdemo:Material')
```

## Response

### Response Status and Error Codes

Code	Description
204	Object type deleted successfully

## 9.1.7 Standard Object Types

There are two standard object types supported:

- **Business partner** - A business partner is an entity with a particular role as an actor in a business scenario. Business partners can be defined on different levels of abstraction, and they can refer to either human beings or to a legal body. Business partner objects can be created using AE APIs. For more details, see the Business Partner section.  
To use standard business partner object type in associations, the entity name should be specified as "com.sap.iotaestandardcontent:BusinessPartner".
- **Location** - Contains the geographical position of a thing. Location objects can be created using Geo Location services or AE APIs. For more details, see the Location section.  
To use standard location object type in associations, the entity name should be specified as "com.sap.iotaestandardcontent:LocationPoint".

### i Note

Use of Geo Location services is recommended.

## 9.2 Object Type Association

Define a master data association between entity types.

### Overview

Association is a way of grouping objects, where every object is related to one or more objects in a unique way. You can create association between entity types, which in turn enables you to create association between object type instances and things. However, the scope of the association between entity types depends on the scope of the package to which the entity types belong.

### Types of object type association

The following are the predefined object type association types:

- onetomany
- onetoone
- manytomany
- manytoone

#### i Note

These are the only values allowed and they are case sensitive.

### Types of relationship

The following are the predefined relationship types:

- *IS A*
- *HAS A*
- *PART OF*

#### i Note

These are the only values allowed and they are case sensitive.

### Types of entities

The following are the predefined entity types:

- *ObjectTypeToObjectType*
- *ObjectTypeToThingType*
- *ThingTypeToObjectType*

#### i Note

These are the only values allowed and they are case sensitive.

**Resource Path:** `://<server address>[:<port number>][<path>]/CustomDataConfiguration`

**Example for a base URI in a cloud foundry environment:** `https://sap-iotaeexplore.iot-sap.cfapps.eu10.hana.ondemand.com/config-customdata-sap/CustomDataConfiguration`

## Operations

### CRUD Operations

HTTP Method	Operation	URI
<i>GET</i>	<a href="#">Read all Object Type Associations [page 627]</a>	/CustomDataConfiguration/v1/ObjectTypeAssociations
<i>POST</i>	<a href="#">Create Object Type Association [page 624]</a>	/CustomDataConfiguration/v1/ObjectTypeAssociations
<i>DELETE</i>	<a href="#">Delete Object Type Association [page 630]</a>	/CustomDataConfiguration/v1/ObjectTypeAssociations(PrimaryEntity='<thing type or Object Type name>',AssociatedEntity='<thing type or object type name>',EntityTypes='Type of Entities associated',Relationship='<relationship name>')

## Entity Type Association Object Properties

Property	Type	Mandatory	Maximum Length	Description
PrimaryEntity	String	Yes	81	Primary entity type for which you want to create an association
AssociatedEntity	String	Yes	81	Entity type associated with the primary entity type
EntityTypes	String	Yes	NA	Association between different types of entities
AssociationType	String	Yes	-	Type of association between the primary entity type and the associated entity type
Relationship	String	Yes	-	Type of relationship between the primary entity type and the associated entity type

Property	Type	Mandatory	Maximum Length	Description
ExpiryTime	TimeStamp	No	-	Expiry time after which the association between the entity types becomes invalid. If you do not specify the expiry time, the association is valid indefinitely.

## Entity Type Association Description Properties

Property	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the object description
Description	String	255	Description of the object

### 9.2.1 Create Object Type Association

With this service, you can create association between entity types, such as object type to object type, object type to thing type and thing type to object type, based on the predefined relationship and association types. The expiry time of the entity type association must be greater than the current time. The association is valid until the defined expiry time, after which, the association is invalid. To extend the validity of the association, you can delete the associated entity and recreate the association with a different expiry time. If you do not specify the expiry time, the association is valid indefinitely.

#### i Note

The primary entity type and the associated entity type must have unique association type and relationship. For example, the entity type **ACBXSeries** is associated with the primary entity type **Truck** with an expiry date **28-10-2018**. The relationship is **IS A** and the association type is **onetoone**. You cannot create another association between the entity types **Truck** and **ACBXSeries** with relationship **IS A** and association type **manytoone**.

#### i Note

It is possible to create more than one type of relationship between two entity types.

## Request

**URI:** /CustomDataConfiguration/v1/ObjectTypeAssociations

**Operation Type:** CRUD

HTTP Method: *POST*

Permissions: <cdconf>.c

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
Name	Yes	String	Unique name of the entity type which represents the primary entity type	Path

## Request Example

/CustomDataConfiguration/v1/ObjectTypeAssociations

## Payload

Format: JSON

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples:

Thing Type to Object Type Association

```
{
  "PrimaryEntity": "iotae.devtest1.cdemo:ConnectedSilo",
  "AssociatedEntity": "iotae.devtest1.cdemo:Material",
  "EntityTypes": "ThingTypeToObjectType",
  "AssociationType": "onetoone",
  "Relationship": "HAS A",
  "ExpiryTime": "2019-10-16T08:26:56.229Z",
  "Descriptions": [
    {
      "LanguageCode": "en",
      "Description": "Connected Silo to Material Association"
    }
  ]
}
```

Object Type To Object Type Association

```
{
  "PrimaryEntity": "iotae.devtest1.cdemo.BP",
  "AssociatedEntity": "iotae.devtest1.cdemo:Material",
  "EntityTypes": "ObjectTypeToObjectType",
  "AssociationType": "onetoone",
  "Relationship": "IS A",
  "ExpiryTime": "2019-10-16T08:26:56.229Z",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "BP to Material Association"
  }]
}
```

## Response

### Response Headers

Header	Description
Etag	Generated Etag; Value: DC68FB05A8AB46619F7F6E22F5F5241F

### Response Status and Error Codes

Code	Description
201	Object type association created successfully.

### Response Payload Example

The following is the response payload for Thing Type to Object Type Association.

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypeAssociations(PrimaryEntity='iotae.devtest1.cdemo
%3AConnectedSilo',AssociatedEntity='iotae.devtest1.cdemo
%3AMaterial',EntityTypes='ThingTypeToObjectType',Relationship='HAS%20A')",
          "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypeAssociations(PrimaryEntity='iotae.devtest1.cdemo
%3AConnectedSilo',AssociatedEntity='iotae.devtest1.cdemo
%3AMaterial',EntityTypes='ThingTypeToObjectType',Relationship='HAS%20A')",
          "type": "com.sap.apptot.ObjectTypeAssociation"
        },
        "EntityTypes": "ThingTypeToObjectType",
        "PrimaryEntity": "iotae.devtest1.cdemo:ConnectedSilo",
        "PrimaryTenant": "iotae-devtest1",
        "AssociatedEntity": "iotae.devtest1.cdemo:Material",
        "Relationship": "HAS A",
        "AssociationExpiryTime": "2019-10-16 08:26:56.229",
        "AssociationCreationTime": "2019-02-15T04:06:50.594Z",
        "AssociatedTenant": "iotae-devtest1",
        "Description": "Connected Silo to Material Association",
        "ETag": null,
        "Descriptions": {
          "__deferred": {
            "uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypeAssociations(PrimaryEntity='iotae.devtest1.cdemo
%3AConnectedSilo',AssociatedEntity='iotae.devtest1.cdemo
%3AMaterial',EntityTypes='ThingTypeToObjectType',Relationship='HAS%20A') /
Descriptions"
          }
        }
      }
    ]
  }
}
```

## 9.2.2 Read all Object Type Associations

With this service, you can retrieve all object type associations.

### Request

**URI:** /CustomDataConfiguration/v1/ObjectTypeAssociations

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <cdconf>.r

### Query Request Parameters

Parameter	Required	Type	Description
\$top	No	Integer	Number of records to include in the result set
\$skip	No	Integer	Number of the first n records to be excluded from the result set  If \$orderby is not chosen, the results will be sorted by the Primary Entity in the ascending order.
\$inlinecount	No	String	Total number of object types available; permissible values are <a href="#">allpages</a> and <a href="#">none</a> . The following results apply if this parameter is used in conjunction with the other query parameters: <ul style="list-style-type: none"><li>• If you use this with the \$top and \$skip parameters, the result contains the total count of the object types available.</li><li>• If you use this with \$filter parameter, the result contains the total count of object types based on the defined filter condition.</li></ul>
\$orderby	No	String	Field name to be used for sorting the result set in ascending order.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.

Parameter	Required	Type	Description
\$filter	No	String	<p>Filter condition to be applied; the only valid fields are PrimaryEntity, AssociatedEntity, Relationship, AssociationType, EntityTypes and ExpiryTime. The filter operators supported are <a href="#">eq</a>, <a href="#">lt</a>, <a href="#">gt</a>, <a href="#">le</a>, <a href="#">ge</a>, and <a href="#">ne</a>.</p> <p>Multiple filters are supported using the separators – <a href="#">and</a>, <a href="#">or</a>.</p> <p>You can use the string functions such as <a href="#">substringof</a>, <a href="#">startswith</a>, and <a href="#">endswith</a> to search data using \$filter. Valid fields are PrimaryEntity, AssociatedEntity, Relationship, AssociationType, EntityTypes and ExpiryTime.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are PrimaryEntity, AssociatedEntity, Relationship, AssociationType, EntityTypes and ExpiryTime</p>

## Request Example

/CustomDataConfiguration/v1/ObjectTypeAssociations

## Response

### Response Status and Error Codes

Code	Description
200	OK

## Response Payload Example

Format: JSON

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://config-customdata-sap-sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/ObjectTypeAssociations(PrimaryEntity='iotae.devtest1.aejob.automobilestes%3AtestOT25',AssociatedEntity='iotae.devtest1.cdatademo%3AMaterial',EntityTypes='ObjectTypeToObjectType',Relationship='HAS%20A')",
          "uri": "https://config-customdata-sap-sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/ObjectTypeAssociations(PrimaryEntity='iotae.devtest1.aejob.automobilestes%3AtestOT25',AssociatedEntity='iotae.devtest1.cdatademo%3AMaterial',EntityTypes='ObjectTypeToObjectType',Relationship='HAS%20A')",
          "type": "com.sap.apptot.ObjectTypeAssociation"
        },

```



```

    "EntityTypes": "ObjectTypeToObjectType",
    "PrimaryEntity": "iotae.devtest1.aejob.automobilestes:testOT25",
    "PrimaryTenant": "iotae-devtest1",
    "AssociatedEntity": "iotae.devtest1.cdatademo:Material",
    "Relationship": "HAS A",
    "AssociationExpiryTime": "2019-10-16T08:26:56Z",
    "AssociationCreationTime": "2019-01-21T12:37:30Z",
    "AssociatedTenant": "iotae-devtest1",
    "Description": "",
    "ETag": "B99EB97AACBB4B6BBEC57B5B638191F3",
    "Descriptions": {
      "_deferred": {
        "_uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypeAssociations(PrimaryEntity='iotae.devtest1.aejob.automobilestes
%3AtestOT25',AssociatedEntity='iotae.devtest1.cdatademo
%3AMaterial',EntityTypes='ObjectTypeToObjectType',Relationship='HAS%20A') /
Descriptions"
      }
    },
    {
      "_metadata": {
        "_id": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypeAssociations(PrimaryEntity='iotae.devtest1.aejob.automobilestes
%3ABP1',AssociatedEntity='iotae.devtest1.aejob.automobilestes
%3AMaterial4',EntityTypes='ObjectTypeToObjectType',Relationship='HAS%20A')",
        "_uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypeAssociations(PrimaryEntity='iotae.devtest1.aejob.automobilestes
%3ABP1',AssociatedEntity='iotae.devtest1.aejob.automobilestes
%3AMaterial4',EntityTypes='ObjectTypeToObjectType',Relationship='HAS%20A')",
        "type": "com.sap.apptot.ObjectTypeAssociation"
      },
      "EntityTypes": "ObjectTypeToObjectType",
      "PrimaryEntity": "iotae.devtest1.aejob.automobilestes:BP1",
      "PrimaryTenant": "iotae-devtest1",
      "AssociatedEntity": "iotae.devtest1.aejob.automobilestes:Material4",
      "Relationship": "HAS A",
      "AssociationExpiryTime": "2030-12-31T23:59:59Z",
      "AssociationCreationTime": "2019-01-18T15:51:57Z",
      "AssociatedTenant": "iotae-devtest1",
      "Description": "",
      "ETag": "67CB398ED2C34E7BABB7FE600B0A6200",
      "Descriptions": {
        "_deferred": {
          "_uri": "https://config-customdata-sap-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataConfiguration/v1/
ObjectTypeAssociations(PrimaryEntity='iotae.devtest1.aejob.automobilestes
%3ABP1',AssociatedEntity='iotae.devtest1.aejob.automobilestes
%3AMaterial4',EntityTypes='ObjectTypeToObjectType',Relationship='HAS%20A') /
Descriptions"
        }
      }
    }
  ]
}
}

```

## 9.2.3 Delete Object Type Association

With this service, you can delete the association between the specified object types.

The `Etag` value is mandatory to delete association of an object type. The `Etag` value is assigned to the request header parameter `If-Match` in the `DELETE` request. If you do not include the `Etag` value in the request header, the server sends HTTP response code 428. If the `Etag` value is not correct, the server sends HTTP response code 412.

You can delete the association between the object types only if an association is not created between the object instances belonging to the associated object types.

### Request

**URI:** `/CustomDataConfiguration/v1/ObjectTypeAssociations(PrimaryEntity='<ThingType or ObjectType name>',AssociatedEntity='<ThingType or ObjectType name>',EntityTypes='Type of Entities associated',Relationship='<relationship name>'`

**Operation Type:** CRUD

**HTTP Method:** `DELETE`

**Permissions:** `<cdconf>.d`

### Request Headers

Header	Required	Values
<code>If-Match</code>	Yes	Latest <code>Etag</code> value of a specific object type association

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
<code>Primary Entity Name</code>	Yes	String	Unique name of the primary thing type/object type to which other thing types are associated	Path
<code>Associated Entity Name</code>	Yes	String	Unique name of the thing type/object type which represents the associated entity	Path
<code>Entity Types</code>	Yes	String	Identify the type of Primary entity and Associated entity. It may have one of the three values: <ul style="list-style-type: none"><li>• <code>ObjectTypeToObjectType</code></li><li>• <code>ThingTypeToObjectType</code></li><li>• <code>ObjectTypeToThingType</code></li></ul>	Path
<code>Relationship</code>	Yes	String	Unique relationship between the associated thing types	Path

## Request Example

/CustomDataConfiguration/v1/ObjectTypeAssociations

(PrimaryEntity='iotae.devtest1.cdemo:ConnectedSilo',AssociatedEntity='iotae.devtest1.cdemo:Material',EntityTypes='ThingTypeToObjectType',Relationship='HAS A')

## Response

### Response Status and Error Codes

Code	Description
204	No Content

## 9.3 Object Type Instances

Define and maintain object type instances and maintain the data of an object type

### Note

- The Master Data Object ID should be validated to be unique per Master Data Object Type within a tenant.

**Resource Path:** `http://<server address>[:<port number>][/path]/ObjectType`

**Example for a base URI in a cloud foundry environment:** `https://sap-iotaeexplore.iot-sap.cfapps.eu10.hana.ondemand.com/customdata/ObjectType`

## Operations

### CRUD Operations

HTTP Method	Operation	URI
GET	<a href="#">Read an Object Type Instance [page 637]</a>	/ObjectType/v1/<FullyQualifiedObjectTypeIDWithUnderscores>('<ObjectInstanceID>')
GET	<a href="#">Read all Object Type Instances [page 638]</a>	/ObjectType/v1/<FullyQualifiedObjectTypeIDWithUnderscores>

HTTP Method	Operation	URI
<i>POST</i>	<a href="#">Create an Object Type Instance [page 634]</a>	/ObjectType/v1/ <FullyQualifiedObjectTypeIDWithUnderscores>
<i>PATCH</i>	<a href="#">Update an Object Type Instance [page 635]</a>	/ObjectType/v1/ <FullyQualifiedObjectTypeIDWithUnderscores>('<ObjectInstanceID>')
<i>DELETE</i>	<a href="#">Delete an Object Type Instance [page 642]</a>	/ObjectType/v1/ <FullyQualifiedObjectTypeIDWithUnderscores>('<ObjectInstanceID>')

#### Custom Operations

HTTP Method	Operation Type	Operation	URI
<i>GET</i>	\$metadata	<a href="#">Metadata of Object Type Instances [page 632]</a>	/ObjectType/v1/\$metadata
<i>POST</i>	\$batch	<a href="#">Metadata of Object Type Instances [page 632]</a>	/ObjectType/v1/\$batch

## 9.3.1 Metadata of Object Type Instances

With this method, you can retrieve the metadata of the specified object type.

### Request

**URI:** /ObjectType/v1/\$metadata

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <cdinstance>.r

### Request Headers

Header	Required	Values
sap-iot-ot	Yes	The Object Type ID for which metadata must be retrieved

Header example: sap-iot-ot:iotae.devtest1.aejob.automobilestes:BusinessPartner

Header example: iotae.devtest1.aejob.automobilestes:Material

### Request Example

/ObjectType/v1/\$metadata

## Response

Format: ATOM

### Response Status and Error Codes

Code	Description
200	OK for Success

### Response Payload Example

```
<?xml version="1.0" ?>
<edmx:Edmx Version="1.0" xmlns:edmx="http://schemas.microsoft.com/ado/2007/06/edmx">
  <edmx:DataServices m:DataServiceVersion="1.0" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:sap="http://www.sap.com/Protocols/SAPData">
    <Schema Namespace="com.sap.appiot" xmlns="http://schemas.microsoft.com/ado/2008/09/edm">
      <EntityType Name="iotae_devtest1_aejob_automobilestes:Material">
        <Documentation>
          <Summary>Details of material desc entity type</Summary>
        </Documentation>
        <Key>
          <PropertyRef Name="ObjectID"></PropertyRef>
        </Key>
        <Property Name="ObjectID" Type="Edm.String" Nullable="false"
MaxLength="36" sap:creatable="true" sap:updatable="false">
          <Documentation>
            <Summary>Object ID of the entity</Summary>
          </Documentation>
        </Property>
        <Property Name="Language" Type="Edm.String" Nullable="false"
MaxLength="20" sap:creatable="true" sap:updatable="false">
          <Documentation>
            <Summary>Language of the entity</Summary>
          </Documentation>
        </Property>
        <Property Name="Density" Type="Edm.String" Nullable="true"
MaxLength="2147483647" sap:creatable="true" sap:updatable="true">
          <Documentation>
            <Summary>Density of entity</Summary>
          </Documentation>
        </Property>
        <Property Name="State" Type="Edm.String" Nullable="true"
MaxLength="2147483647" sap:creatable="true" sap:updatable="true">
          <Documentation>
            <Summary>State of entity</Summary>
          </Documentation>
        </Property>
      </EntityType>
      <EntityContainer Name="ObjectTypeInstanceContainer"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="iotae_devtest1_aejob_automobilestes:Material"
EntityType="com.sap.appiot.iotae_devtest1_aejob_automobilestes:Material">
          <Documentation>
            <Summary>Details of material desc entity set</Summary>
          </Documentation>
        </EntitySet>
      </EntityContainer>
    </Schema>
  </edmx:DataServices>
</edmx:Edmx>
```

## 9.3.2 Create an Object Type Instance

With this method, you can create an object instance for the object type.

### i Note

- For all the CRUD operations, the fully qualified Object Type ID in the URL should not contain dots(.). All the dots should be replaced with underscores. Example: Object Type `iotae.devtest1.aejob:automobilestes:BusinessPartner` has to be passed as `iotae_devtest1_aejob_automobilestes:BusinessPartner`

## Request

**URI:** `/ObjectType/v1/<FullyQualifiedObjectTypeIDWithUnderscores>`

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** `<cdinstance>.c`

### Request Headers

Header	Required	Values
Content-Type	Yes	The content type of the payload

Header example: `Content-Type: application/json`

### Request Example

`/ObjectType/v1/iotae_devtest1_aejob_automobilestes:Material`

## Payload

**Format:** JSON

### i Note

Object ID: User can provide this ID or the system will generate a unique GUID.

```
{
  "ObjectID": "MAT-01",
  "Density": "{ \"en\": \"200\", \"fr\": \"200\" }",
  "State": "Solid"
}
```

```
}
```

## Response

Format: ATOM or JSON

### Response Status and Error Codes

Code	Description
201	Object type instance created successfully

### Response Payload Example

```
<?xml version="1.0" encoding="utf-8"?>
<entry xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://
schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://
schemas.microsoft.com/ado/2007/08/dataservices" xml:base="http://localhost:9080/
customdata-services/ObjectType/v1/">
  <id>http://localhost:9080/customdata-services/ObjectType/v1/
iotae_devtest1_aejob_automobilestes%3AMaterial('MAT-01')</id>
  <title type="text">iotae_devtest1_aejob_automobilestes:Material</title>
  <updated>2019-01-07T12:16:04.417+05:30</updated>
  <category term="com.sap.appiot.iotae_devtest1_aejob_automobilestes:Material"
scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category>
  <link href="iotae_devtest1_aejob_automobilestes%3AMaterial('MAT-01') "
rel="edit" title="iotae_devtest1_aejob_automobilestes:Material"></link>
  <content type="application/xml">
    <m:properties>
      <d:ObjectID>MAT-01</d:ObjectID>
      <d:State>Solid</d:State>
      <d:Density>{"en":"200","fr":"200"}</d:Density>
    </m:properties>
  </content>
</entry>
```

## 9.3.3 Update an Object Type Instance

With this method, you can update an object instance for the object type.

### i Note

- For all the CRUD operations, the fully qualified Object Type ID in the URL should not contain dots(.). All the dots should be replaced with underscores. Example: Object Type `iotae_devtest1_aejob:automobilestes:BusinessPartner` has to be passed as `iotae_devtest1_aejob_automobilestes:BusinessPartner`
- You cannot update an object ID using a PATCH request. The object ID specified in the request payload is ignored for update.

## Request

**URI:** /ObjectType/v1/<FullyQualifiedObjectTypeIDWithUnderscores>('<ObjectInstanceID>')

**Operation Type:** CRUD

**HTTP Method:** [\[PATCH\]](#)

**Permissions:** <cdinstance>.u

### Request Headers

Header	Required	Values
Content-Type	Yes	The content type of the payload

Header example: Content-Type: application/json

### Request Parameters

Parameter	Required	Data Type	Description
ObjectInstanceID	Yes	String	Unique identifier which was auto-generated/specified while creating the object type instance

### Request Example

```
/ObjectType/v1/iotae_devtest1_aejob_automobilestes:Material('MAT-01')
```

## Payload

```
{
  "Density": "{ \"en\": \"400\", \"de\": \"400\", \"fr\": \"400\" }",
  "State": "Solid"
}
```

## Response

### Response Status and Error Codes

Code	Description
204	No content for success



## 9.3.4 Read an Object Type Instance

With this method, you can get a single object instance for the object type.

### i Note

- For all the CRUD operations, the fully qualified Object Type ID in the URL should not contain dots(.). All the dots should be replaced with underscores. Example: Object Type `iotae.devtest1.aejob:automobilestes:BusinessPartner` has to be passed as `iotae_devtest1_aejob_automobilestes:BusinessPartner`

## Request

**URI:** `/ObjectType/v1/<FullyQualifiedObjectTypeIDWithUnderscores>('<ObjectInstanceID>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<cdinstance>.r`

### Request Parameters

Parameter	Required	Data Type	Description
ObjectInstanceID	Yes	String	Unique identifier which was auto-generated/specified while creating the object type instance

### Request Header

Name	Required	Data Type	Description
Accept-Language	Optional	string	Accepted language header.

### i Note

Based on the choice of language, the values will be returned..

### Request Example

`/ObjectType/v1/iotae_devtest1_aejob_automobilestes:Material('Material1')`

## Response

Format: ATOM/JSON

## Response Status and Error Codes

Code	Description
200	OK for Success

## Response Payload Example

```
<?xml version="1.0" encoding="utf-8"?>
<entry xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://
schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://
schemas.microsoft.com/ado/2007/08/dataservices" xml:base="http://localhost:9080/
customdata-services/ObjectType/v1/">
  <id>http://localhost:9080/customdata-services/ObjectType/v1/
iotaedevtest1aejobautomobilestes%3AMaterial('Material1')</id>
  <title type="text">iotaedevtest1aejobautomobilestes:Material</title>
  <updated>2019-01-07T11:46:11.05+05:30</updated>
  <category term="com.sap.apptot.iotaedevtest1aejobautomobilestes:Material"
scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category>
  <link href="iotaedevtest1aejobautomobilestes%3AMaterial('Material1') "
rel="edit" title="iotaedevtest1aejobautomobilestes:Material"></link>
  <content type="application/xml">
    <m:properties>
      <d:ObjectID>Material1</d:ObjectID>
      <d:Density>{"EN\":"300"}</d:Density>
      <d:State>{"EN\":"null"}</d:State>
    </m:properties>
  </content>
</entry>[response example]
```

## 9.3.5 Read all Object Type Instances

With this method, you can get all the object instances for an object type.

### i Note

- For all the CRUD operations, the fully qualified Object Type ID in the URL should not contain dots(.). All the dots should be replaced with underscores. Example: Object Type `iotaedevtest1aejobautomobilestes:BusinessPartner` has to be passed as `iotaedevtest1aejobautomobilestes:BusinessPartner`

## Request

**URI:** /ObjectType/v1/<FullyQualifiedObjectTypeIDWithUnderscores>

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <cdinstance>.r

## Query Request Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set
\$skip	No	Integer	Number of the first n records to be excluded from the result set
\$inlinecount	No	String	<p>Total number of object types available; permissible values are <i>allpages</i> and <i>none</i>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"> <li>If you use this with the \$top and \$skip parameters, the result contains the total count of the object types available.</li> <li>If you use this with \$filter parameter, the result contains the total count of the object types based on the defined filter condition.</li> </ul>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order.</p> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; only valid fields are ObjectID and dynamic fields defined for the object type. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the separators – <i>and</i>, <i>or</i>.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using \$filter. The valid fields are ObjectID and dynamic fields defined for the object type.</p>
\$select	No	String	Select condition for the data fields to be included in the result set; the only valid fields are ObjectID and dynamic fields defined for the object type.

## Request Header

Name	Required	Data Type	Description
Accept-Language	Optional	string	Accepted language header.

### Note

Based on the choice of language, the values will be returned.

## Request Example

```
/ObjectType/v1/iotae_devtest1_aejob_automobilestes:Material
```

## Response

Format: ATOM/JSON

## Response Status and Error Codes

Code	Description
200	OK for Success

## Response Payload Example

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('Material1')",
          "uri": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('Material1')",
          "type":
"com.sap.apptot.iotae_devtest1_aejob_automobilestes:Material"
        },
        "ObjectID": "Material1",
        "Density": "{\\\\"EN\\\\":\\\\"300\\\\"}",
        "State": "{\\\\"EN\\\\":\\\\"null\\\\"}"
      },
      {
        "__metadata": {
          "id": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('ABCD1234')",
          "uri": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('ABCD1234')",
          "type":
"com.sap.apptot.iotae_devtest1_aejob_automobilestes:Material"
        },
        "ObjectID": "ABCD1234",
        "Density": "{\\\\"de\\\\":\\\\"300\\\\"", \\\\"en\\\\":\\\\"400\\\\"}",
        "State": "{\\\\"de\\\\":\\\\"null\\\\"", \\\\"en\\\\":\\\\"null\\\\"}"
      }
    ]
  }
}
```

```

    },
    {
      "__metadata": {
        "id": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P1')",
        "uri": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P1')",
        "type":
"com.sap.appiot.iotae_devtest1_aejob_automobilestes:Material"
      },
      "ObjectID": "P1",
      "Density": "{\\\\"en\\\\":\\\\"null\\\\"}",
      "State": "{\\\\"en\\\\":\\\\"null\\\\"}"
    },
    {
      "__metadata": {
        "id": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P3')",
        "uri": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P3')",
        "type":
"com.sap.appiot.iotae_devtest1_aejob_automobilestes:Material"
      },
      "ObjectID": "P3",
      "Density": "{\\\\"en\\\\":\\\\"null\\\\"", \\\\"de\\\\":\\\\"400\\\\"}",
      "State": "{\\\\"en\\\\":\\\\"null\\\\"", \\\\"de\\\\":\\\\"null\\\\"}"
    },
    {
      "__metadata": {
        "id": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P4')",
        "uri": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P4')",
        "type":
"com.sap.appiot.iotae_devtest1_aejob_automobilestes:Material"
      },
      "ObjectID": "P4",
      "Density": "{\\\\"de\\\\":\\\\"200\\\\"", \\\\"en\\\\":\\\\"null\\\\"", \\
\\\"fr\\\\":\\\\"200\\\\"}",
      "State": "{\\\\"de\\\\":\\\\"null\\\\"", \\\\"en\\\\":\\\\"Solid\\\\"", \\
\\\"fr\\\\":\\\\"null\\\\"}"
    },
    {
      "__metadata": {
        "id": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('Material2')",
        "uri": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('Material2')",
        "type":
"com.sap.appiot.iotae_devtest1_aejob_automobilestes:Material"
      },
      "ObjectID": "Material2",
      "Density": "{\\\\"en\\\\":\\\\"500\\\\"", \\\\"de\\\\":\\\\"600\\\\"}",
      "State": "{\\\\"en\\\\":\\\\"null\\\\"", \\\\"de\\\\":\\\\"null\\\\"}"
    },
    {
      "__metadata": {
        "id": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('Material3')",
        "uri": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('Material3')",
        "type":
"com.sap.appiot.iotae_devtest1_aejob_automobilestes:Material"
      },
      "ObjectID": "Material3",
      "Density": "{\\\\"de\\\\":\\\\"900\\\\"}",
      "State": "{\\\\"de\\\\":\\\\"null\\\\"}"
    },
  ],

```

```

{
  "__metadata": {
    "id": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P5')",
    "uri": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P5')",
    "type":
"com.sap.apptot.iotae_devtest1_aejob_automobilestes:Material"
  },
  "ObjectID": "P5",
  "Density": "{\\\\"de\\\\":\\\\"200\\\\", \\\\"en\\\\":\\\\"null\\\\"", \\
\\fr\\\\":\\\\"200\\\\"}",
  "State": "{\\\\"de\\\\":\\\\"null\\\\"", \\\\"en\\\\":\\\\"Solid\\\\"", \\
\\fr\\\\":\\\\"null\\\\"}"
},
{
  "__metadata": {
    "id": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P6')",
    "uri": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P6')",
    "type":
"com.sap.apptot.iotae_devtest1_aejob_automobilestes:Material"
  },
  "ObjectID": "P6",
  "Density": "{\\\\"de\\\\":\\\\"200\\\\"", \\\\"en\\\\":\\\\"null\\\\"", \\
\\fr\\\\":\\\\"200\\\\"}",
  "State": "{\\\\"de\\\\":\\\\"null\\\\"", \\\\"en\\\\":\\\\"Solid\\\\"", \\
\\fr\\\\":\\\\"null\\\\"}"
},
{
  "__metadata": {
    "id": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P7')",
    "uri": "http://localhost:9080/customdata-services/
ObjectType/v1/iotae_devtest1_aejob_automobilestes%3AMaterial('P7')",
    "type":
"com.sap.apptot.iotae_devtest1_aejob_automobilestes:Material"
  },
  "ObjectID": "P7",
  "Density": "{\\\\"de\\\\":\\\\"200\\\\"", \\\\"en\\\\":\\\\"null\\\\"}",
  "State": "{\\\\"de\\\\":\\\\"null\\\\"", \\\\"en\\\\":\\\\"Solid\\\\"}"
}
]
}
}

```

## 9.3.6 Delete an Object Type Instance

With this method, you can delete an object instance of the object type.

### i Note

- For all the CRUD operations, the fully qualified Object Type ID in the URL should not contain dots(.). All the dots should be replaced with underscores. Example: Object Type `iotae.devtest1.aejob:automobilestes:BusinessPartner` has to be passed as `iotae_devtest1_aejob_automobilestes:BusinessPartner`

## Request

**URI:** /ObjectType/v1/<FullyQualifiedObjectTypeIDWithUnderscores>('<ObjectInstanceID>')

**Operation Type:** CRUD

**HTTP Method:** *DELETE*

**Permissions:** <cdinstance>.d

### Request Parameters

Parameter	Required	Data Type	Description
ObjectInstanceID	Yes	String	Unique identifier which was auto-generated/specified while creating the object type instance

### Request Example

```
/ObjectType/v1/iotae_devtest1_aejob_automobilestes:Material('Material1')
```

## Response

### Response Status and Error Codes

Code	Description
204	No content for success

## 9.3.7 \$Batch

With this API, you can batch multiple calls in a single request.

### i Note

Each batch request URL has to be encoded. For example:

Single request: /iotae\_devtest1\_cdemo:Material

Batch: /iotae\_devtest1\_cdemo%3AMaterial

## Request

**URI:** /ObjectType/v1/\$batch

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** <cdinstance>.c,<cdinstance>.r,<cdinstance>.d,<cdinstance>.u

## Request Headers

Header	Required	Values
sap-iot-ot	Yes	The Object Type ID for which \$batch requests need to be executed Example: iotae.devtest1.cdatademo:Material
Content-Type	Yes	multipart/mixed;batchboundary=<batch boundary> Example: multipart/mixed;boundary=batch_mybatch

## Request Example

POST /ObjectType/v1/\$batch

## Payload

```
--batch_mybatch
Content-Type: application/http
Content-Transfer-Encoding:binary

GET iotae_devtest1_cdatademo%3AMaterial('0A444FC08A874E8DB6EFDBDE1D6C656C') HTTP/
1.1
sap-contextid-accept: header
Accept: application/json
Accept-Language: en
DataServiceVersion: 2.0
MaxDataServiceVersion: 2.0

--batch_mybatch
Content-Type: multipart/mixed; boundary=changeset_c1

--changeset_c1
Content-Type: application/http
Content-Transfer-Encoding: binary

POST iotae_devtest1_cdatademo%3AMaterial HTTP/1.1
Content-Type: application/json
Accept: application/json

{
  "MATNR":"A25"
}

--changeset_c1
Content-Type: application/http
Content-Transfer-Encoding:binary

PATCH iotae_devtest1_cdatademo%3AMaterial('0A444FC08A874E8DB6EFDBDE1D6C656C')
HTTP/1.1
Content-Type: application/json
Accept: application/json

{
```



```
"MATNR":"A26"
}

--changeset_c1--
--batch_mybatch--
```

## Response

### Response Status and Error Codes

Code	Description
202	Accepted

### Response Payload Example

```
--batch_863607c3-f1ea-47d9-87a7-e759c4e31740
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 200 OK
DataServiceVersion: 2.0
Content-Type: application/json
Content-Length: 498

{"d":{"__metadata":{"id":"https://customdata-sandbox.cfapps.sap.hana.ondemand.com:443/ObjectType/v1/iotae_devtest1_cdatademo%3AMaterial('0A444FC08A874E8DB6EFDBDE1D6C656C')","uri":"https://customdata-sandbox.cfapps.sap.hana.ondemand.com:443/ObjectType/v1/iotae_devtest1_cdatademo%3AMaterial('0A444FC08A874E8DB6EFDBDE1D6C656C')","type":"com.sap.apptot.iotae_devtest1_cdatademo:Material"},"ObjectID":"0A444FC08A874E8DB6EFDBDE1D6C656C","MATNR":"A26","Material_Description":{"\\\\"en\\\\"::\\"Sugar\\\\"}}}}
--batch_863607c3-f1ea-47d9-87a7-e759c4e31740
Content-Type: multipart/mixed;
boundary=changeset_382827d0-9d10-46a6-92cd-3ed5f5ce7dd2

--changeset_382827d0-9d10-46a6-92cd-3ed5f5ce7dd2
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 201 Created
DataServiceVersion: 2.0
Location: v1/iotae_devtest1_cdatademo:Material('C7F5160D36CA4EF892D6699F20C7E2A5')
Content-Type: application/json
Content-Length: 447

{"d":{"__metadata":{"id":"https://customdata-sandbox.cfapps.sap.hana.ondemand.com:443/ObjectType/v1/iotae_devtest1_cdatademo%3AMaterial('C7F5160D36CA4EF892D6699F20C7E2A5')","uri":"https://customdata-sandbox.cfapps.sap.hana.ondemand.com:443/ObjectType/v1/iotae_devtest1_cdatademo%3AMaterial('C7F5160D36CA4EF892D6699F20C7E2A5')","type":"com.sap.apptot.iotae_devtest1_cdatademo:Material"},"ObjectID":"C7F5160D36CA4EF892D6699F20C7E2A5","MATNR":"A25"}}}
--changeset_382827d0-9d10-46a6-92cd-3ed5f5ce7dd2
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 204 No Content
DataServiceVersion: 2.0
```

```
--changeset_382827d0-9d10-46a6-92cd-3ed5f5ce7dd2--  
--batch_863607c3-f1ea-47d9-87a7-e759c4e31740--
```

## 9.4 Instance Associations

You can associate one or more object type instances to other object type instances or thing instances. This is possible only if an association already exists between the object types or the object type and the thing type respectively.

**Resource Path:** `http://<server address>[:<port number>][/path]/CustomDataAssociation`

**Example for a base URI in a cloud foundry environment:** `https://sap-iotaeexplore.iot-sap.cfapps.eu10.hana.ondemand.com/customdata-associations/CustomDataAssociation`

### Operations

#### CRUD Operations

HTTP Method	Operation	URI
<i>GET</i>	<a href="#">Read an Instance Association [page 649]</a>	<code>/CustomDataAssociation/v1/ObjectAssociations('&lt;Association ID&gt;')</code>
<i>GET</i>	<a href="#">Read all Instance Associations [page 650]</a>	<code>/CustomDataAssociation/v1/ObjectAssociations</code>
<i>POST</i>	<a href="#">Create an Instance Association [page 647]</a>	<code>/CustomDataAssociation/v1/ObjectAssociations</code>
<i>DELETE</i>	<a href="#">Delete an Instance Association [page 652]</a>	<code>/CustomDataAssociation/v1/ObjectAssociations('&lt;Association ID&gt;')</code>

#### Custom Operations

HTTP Method	Operation Type	Operation	URI
<i>GET</i>	\$metadata		<code>/v1/\$metadata</code>
<i>POST</i>	\$batch	<a href="#">\$Batch [page 653]</a>	<code>/CustomDataAssociation/v1/\$batch</code>

## 9.4.1 Create an Instance Association

With this method, you can create an object instance association.

### i Note

Only one association can be created in a POST request.

### Request

**URI:** /CustomDataAssociation/v1/ObjectAssociations

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** <thing>.c

### Request Example

### Payload

**Format:** JSON

/CustomDataAssociation/v1/ObjectAssociations

```
{
  "PrimaryObjectID": "BP1",
  "AssociatedObjectID": "MAT1",
  "PrimaryEntity": "iotae.devtest1.aejob.automobilestes:BP1",
  "AssociatedEntity": "iotae.devtest1.aejob.automobilestes:MAT1",
  "EntityTypes": "ObjectTypeToObjectType",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Association between Partner BP1 and Material MAT1"
  }],
  "Relationship": "HAS A",
  "StartTime": "2019-05-17T13:00:56.229Z",
  "ExpiryTime": "2019-06-17T12:00:56.229Z"
}
```

Accepted EntityTypes

- ThingTypeToObjectType - To associate Thing type to Object type
- ObjectTypeToThingType - To associate Object type to Thing type
- ObjectTypeToObjectType - To associate Object type to Thing type

Accepted Relationship

- IS A
- HAS A
- PART OF

## Response

### Response Headers

Header	Description
Etag	87FF06A3B6784A3B9F4A6EC3F9EA0675

### Response Status and Error Codes

Code	Description
201	Object association created successfully

### Response Payload Example

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://customdata-associations-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataAssociation/v1/
ObjectAssociations('E0E33B95BCB1490AB7C50A87C6C69988')",
          "uri": "https://customdata-associations-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataAssociation/v1/
ObjectAssociations('E0E33B95BCB1490AB7C50A87C6C69988')",
          "type": "com.sap.apptot.ObjectAssociation"
        },
        "AssociationID": "E0E33B95BCB1490AB7C50A87C6C69988",
        "PrimaryObjectID": "BP1",
        "AssociatedObjectID": "MAT1",
        "PrimaryEntity": "iotae.devtest1.aejob.automobilestes:BP1",
        "AssociatedEntity": "iotae.devtest1.aejob.automobilestes:MAT1",
        "EntityTypes": "ObjectTypeToObjectType",
        "Relationship": "HAS A",
        "StartTime": "2019-05-17T13:00:56.229Z",
        "ExpiryTime": "2019-06-17T12:00:56.229Z",
        "CreationTime": "2019-03-06T09:23:08.195Z",
        "Description": "Association between Partner BP1 and Material
MAT1",
        "Descriptions": {
          "__deferred": {
            "uri": "https://customdata-associations-
sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataAssociation/v1/
ObjectAssociations('E0E33B95BCB1490AB7C50A87C6C69988')/Descriptions"
          }
        }
      }
    ]
  }
}
```

## 9.4.2 Read an Instance Association

With this method, you can read the details of an object instance association.

### Request

**URI:** /CustomDataAssociation/v1/ObjectAssociations('<Association ID>')

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <thing>.r

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
Association ID	Yes	String	Unique identifier of the object association	Path

### Response

#### Response Status and Error Codes

Code	Description
200	OK

#### Response Payload Example

**Format:** JSON

/CustomDataAssociation/v1/ObjectAssociations('CF42E985C3104789BD128CBC8A6B996B')

```
{
  "d": {
    "results": [{
      "AssociationId": "CF42E985C3104789BD128CBC8A6B996B",
      "PrimaryObjectID": "4B90908372BB4648BEA1EFDD56B0819C",
      "AssociatedObjectID": "72136789129",
      "PrimaryEntity": "core.automobiles:ABCXSeries",
      "AssociatedEntity": "core.automobiles:BusinessPartner",
      "EntityTypes": "ThingTypeToObjectType",
      "Description": "Association between Partner DEF and ABCXSeries1",
      "Relationship": "IS A",
      "StartTime": "2019-01-01T08:26:56.229Z",
      "ExpiryTime": "2019-05-31T08:26:56.229Z"
    }]
  }
}
```

## 9.4.3 Read all Instance Associations

With this method, you can read all object instance associations which are valid at the current time. If you want to read any past or future associations, you can use the StartTime/ExpiryTime filter.

### Request

**URI:** /CustomDataAssociation/v1/ObjectAssociations

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <thing>.r

### Query Request Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set
\$skip	No	Integer	Number of the first n records to be excluded from the result set
\$inlinecount	No	String	Total number of object types available; permissible values are <a href="#">allpages</a> and <a href="#">none</a> . The following results apply if this parameter is used in conjunction with the other query parameters: <ul style="list-style-type: none"><li>• If you use this with the \$top and \$skip parameters, the result contains the total count of the object types available.</li><li>• If you use this with \$filter parameter, the result contains the total count of the object types based on the defined filter condition.</li></ul>
\$orderby	No	String	Field name to be used for sorting the result set in ascending order.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.

Parameter	Required	Data Type	Description
\$filter	No	String	<p>Filter condition to be applied; only valid fields are AssociationID, PrimaryObjectID, PrimaryEntity, AssociatedObjectID, AssociatedEntity, Relationship, StartTime and ExpiryTime. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the separators – <i>and</i>, <i>or</i>.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using \$filter. Only valid fields are AssociationID, PrimaryObjectID, PrimaryEntity, AssociatedObjectID, AssociatedEntity, Relationship, StartTime and ExpiryTime.</p>
\$select	No	String	Select condition for the data fields to be included in the result set; only valid fields are AssociationID, PrimaryObjectID, PrimaryEntity, AssociatedObjectID, AssociatedEntity, Relationship, StartTime and ExpiryTime.

## Response

### Response Status and Error Codes

Code	Description
200	OK

### Response Payload Example

Format: JSON

/CustomDataAssociation/v1/ObjectAssociations

```
{
  "d": {
    "results": [{
      "AssociationId": "CF42E985C3104789BD128CBC8A6B996B",
      "PrimaryObjectID": "4B90908372BB4648BEA1EFDD56B0819C",
      "AssociatedObjectID": "72136789129",
      "PrimaryEntity": "core.automobiles:ABCXSeries",
      "AssociatedEntity": "core.automobiles:BusinessPartner",
      "EntityTypes": "ThingTypeToObjectType",
      "Description": "Association between Partner DEF and ABCXSeries1",
      "Relationship": "IS A",
      "StartTime": "2019-01-01T08:26:56.229Z",
      "ExpiryTime": "2019-05-31T08:26:56.229Z"
    },
    {
      "AssociationId": "CF42E985C3104789BD128CBC8A6C666B",
      "PrimaryObjectID": "4B90908372BB4648BEA1EFDD56B0819C",
      "AssociatedObjectID": "72136789130",
    }
  ]
}
```

```

    "PrimaryEntity": "core.automobiles:ABCXSeries",
    "AssociatedEntity": "core.automobiles:BusinessPartner",
    "EntityTypes": "ThingTypeToObjectType",
    "Description": "Association between Partner DEF and ABCXSeries1",
    "Relationship": "IS A",
    "StartTime": "2019-07-01T08:26:56.229Z",
    "ExpiryTime": "2019-12-31T08:26:56.229Z"
  }
}
}

```

## 9.4.4 Delete an Instance Association

With this method, you can delete an object instance association.

### Request

**URI:** /CustomDataAssociation/v1/ObjectAssociations('<Association ID>')

**Operation Type:** CRUD

**HTTP Method:** *DELETE*

**Permissions:** <thing>.d

### Request Headers

Header	Required	Values
If-Match	Yes	Latest Etag value of the specified object instance association

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
Association ID	Yes	String	Unique identifier of the object association	Path

### Request Example

```

/CustomDataAssociation/v1/ObjectAssociations('CF42E985C3104789BD128CBC8A6B996B')

```



## Response

### Response Status and Error Codes

Code	Description
204	No content

## 9.4.5 \$Batch

With this API, you can batch multiple calls in a single request.

### i Note

Request with `Content-ID` is not supported.

## Request

**URI:** `/CustomDataAssociation/v1/$batch`

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** `<thing>.c, <thing>.r, <thing>.d`

### Request Headers

Header	Required	Values
Content-Type	Yes	multipart/mixed;batchboundary=<batch boundary> Example: multipart/mixed;boundary=batch_mybatch

### Request Example

Request Header `Content-Type: multipart/mixed;boundary=batch_mybatch`

## Payload

```
--batch_mybatch
Content-Type: application/http
Content-Transfer-Encoding:binary

GET ObjectAssociations('4C95D612A2BE43C5BF4DC145FCADD89D') HTTP/1.1
Accept: application/json
```

```

Accept-Language: en
DataServiceVersion: 2.0
MaxDataServiceVersion: 2.0

--batch_mybatch
Content-Type: multipart/mixed; boundary=changeset_c1

--changeset_c1
Content-Type: application/http
Content-Transfer-Encoding: binary

POST ObjectAssociations HTTP/1.1
Content-Type: application/json

{
  "PrimaryObjectID": "BP1",
  "AssociatedObjectID": "MAT1",
  "PrimaryEntity": "iotae.devtest1.aejob.automobilestes:BP1",
  "AssociatedEntity": "iotae.devtest1.aejob.automobilestes:MAT1",
  "EntityTypes": "ObjectTypeToObjectType",
  "Descriptions": [
    {
      "LanguageCode": "en",
      "Description": "Association between Partner BP1 and Material MAT1"
    }
  ],
  "Relationship": "HAS A",
  "StartTime" : "2019-08-18T13:00:56.229Z",
  "ExpiryTime": "2019-08-19T12:00:56.229Z"
}

--changeset_c1--
--batch_mybatch--

```

## Response

### Response Status and Error Codes

Code	Description
202	Accepted

### Response Payload Example

```

--batch_dee143ca-c84b-4e1b-bad1-c023ba91c202
Content-Type: application/http
Content-Transfer-Encoding: binary

HTTP/1.1 200 OK
DataServiceVersion: 2.0
Content-Type: application/json
Content-Length: 1025

{"d":{"__metadata":{"id":"https://customdata-associations-sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataAssociation/v1/ObjectAssociations('4C95D612A2BE43C5BF4DC145FCADD89D')","uri":"https://customdata-associations-sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataAssociation/v1/ObjectAssociations('4C95D612A2BE43C5BF4DC145FCADD89D')","type":"com.sap.apptot.ObjectAssociation"},"AssociationID":"4C95D612A2BE43C5BF4DC145FCADD89D","PrimaryObj

```

```

ctID":"P12","AssociatedObjectID":"P56","PrimaryEntity":"iotae.devtest1.aejob.automobilestes:Material","AssociatedEntity":"iotae.devtest1.aejob.automobilestes:Material3","EntityTypes":"ObjectTypeToObjectType","Relationship":"HAS A","StartTime":"2019-01-17T18:03:50.000Z","ExpiryTime":"2020-12-31T23:59:59.000Z","CreationTime":"2019-01-17T18:04:39.000Z","Description":null,"Descriptions":{"__deferred":{"uri":"https://customdata-associations-sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataAssociation/v1/ObjectAssociations('4C95D612A2BE43C5BF4DC145FCADD89D')/Descriptions"}}}
--batch_deel43ca-c84b-4e1b-bad1-c023ba91c202
Content-Type: multipart/mixed; boundary=changeset_48bf8927-353b-4998-a6cf-9348c0718321

```

```

--changeset_48bf8927-353b-4998-a6cf-9348c0718321
Content-Type: application/http
Content-Transfer-Encoding: binary

```

```

HTTP/1.1 201 Created
ETag: 1700F7719CCA4C14A4CE854C050ED883
DataServiceVersion: 2.0
Location: v1/ObjectAssociations('2B94D029175B466ABC77C5480293DB06')
Content-Type: application/atom+xml;charset=utf-8;type=entry
Content-Length: 2105

```

```

<?xml version="1.0" encoding="utf-8"?><feed xmlns="http://www.w3.org/2005/Atom"
xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata"
xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices"
xml:base="https://customdata-associations-sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataAssociation/v1/"><id>https://customdata-associations-sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataAssociation/v1/ObjectAssociations</id><title type="text">ObjectAssociations</title><updated>2019-02-18T02:57:21.825Z</updated><author><name></name></author><link href="ObjectAssociations" rel="self" title="ObjectAssociations"></link><entry><id>https://customdata-associations-sandbox.cfapps.sap.hana.ondemand.com:443/CustomDataAssociation/v1/ObjectAssociations('2B94D029175B466ABC77C5480293DB06')</id><title type="text">ObjectAssociations</title><updated>2019-02-18T02:57:21.828Z</updated><category term="com.sap.apptot.ObjectAssociation" scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category><link href="ObjectAssociations('2B94D029175B466ABC77C5480293DB06')> rel="edit" title="ObjectAssociation"></link><link href="ObjectAssociations('2B94D029175B466ABC77C5480293DB06')/Descriptions" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/Descriptions" title="Descriptions" type="application/atom+xml;type=feed"></link><content type="application/xml"><m:properties><d:AssociationID>2B94D029175B466ABC77C5480293DB06</d:AssociationID><d:PrimaryObjectID>BP1</d:PrimaryObjectID><d:AssociatedObjectID>MAT1</d:AssociatedObjectID><d:PrimaryEntity>iotae.devtest1.aejob.automobilestes:BP1</d:PrimaryEntity><d:AssociatedEntity>iotae.devtest1.aejob.automobilestes:MAT1</d:AssociatedEntity><d:EntityTypes>ObjectTypeToObjectType</d:EntityTypes><d:Relationship>HAS A</d:Relationship><d:StartTime>2019-08-18T13:00:56.229Z</d:StartTime><d:ExpiryTime>2019-08-19T12:00:56.229Z</d:ExpiryTime><d:CreationTime>2019-02-18T02:57:21.788Z</d:CreationTime><d:Description>Association between Partner BP1 and Material MAT1</d:Description></m:properties></content></entry></feed>
--changeset_48bf8927-353b-4998-a6cf-9348c0718321--
--batch_deel43ca-c84b-4e1b-bad1-c023ba91c202--

```

## 9.4.6 Create Instance Association with Standard Object Types

To create instance association with Standard Object Type Business Partner or Location, the same create instance association API must be used, specifying the standard object type name in the PrimaryEntity/AssociatedEntity.

### Request

**URI:** /CustomDataAssociation/v1/ObjectAssociations

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** <thing>.c

**Request Example (Create instance association with Location standard object type)**

### Payload

**Format:** JSON

/CustomDataAssociation/v1/ObjectAssociations

```
{
  "PrimaryObjectID": "BP1",
  "AssociatedObjectID": "LOC1",
  "PrimaryEntity": "iota.devtest1.aejob.automobilestes:BP1",
  "AssociatedEntity": "com.sap.iotaestandardcontent:LocationPoint",
  "EntityTypes": "ObjectTypeToObjectType",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description": "Association between Partner BP1 and Location LOC1"
  }],
  "Relationship": "HAS A",
  "StartTime": "2019-05-17T13:00:56.229Z",
  "ExpiryTime": "2019-06-17T12:00:56.229Z"
}
```

## 9.5 Role Templates and Scopes

Standard role templates provided by SAP for typical user profiles

### Overview

The big number of different scopes that are needed to grant (or revoke) functional authorizations in SAP Leonardo IoT can turn into a challenge for administrators who are in charge of assigning proper user profiles for the users of the system. To make this task easier, SAP provides a number of predefined role templates that an administrator may use as a blueprint for setting up individual roles as they are needed for a concrete organization. The basic procedure for this is as follows:

1. Derive a role from a role template.
2. Assign role (one or many) to a role collection.
3. Assign role collection (one or many) to a user group.

Role templates and scopes have a n:m relationship. That is, one template may contain many scopes, and one scope may be contained in many templates. The following sections show listings of the available role templates and the scopes that are contained in each template. The listings are broken down by the main areas of the SAP IoT Application Services platform.

### Master Data

Role Templates for Master Data

Role Template	Description	Scopes
Masterdata_Viewer	View master data	<cdconf>.r  <cdinstance>.r

Role Template	Description	Scopes
Masterdata_Editor	Edit master data <ul style="list-style-type: none"> <li>Read ObjectTypes</li> <li>Create ObjectTypes</li> <li>Delete ObjectTypes</li> <li>Update ObjectTypes</li> <li>Read Instances</li> <li>Create instances</li> <li>Delete instances</li> <li>Update instances</li> <li>Read ObjectTypeAssociations</li> <li>Create ObjectTypeAssociations</li> <li>Delete ObjectTypeAssociations</li> </ul>	<cdconf>.r <cdconf>.c <cdconf>.d <cdconf>.u <cdinstance>.r <cdinstance>.c <cdinstance>.d <cdinstance>.u <cdconf>.r <cdconf>.c <cdconf>.d
Thing_Engineer	Thing Engineer <ul style="list-style-type: none"> <li>Read ObjectTypes</li> <li>Create ObjectTypes</li> <li>Delete ObjectTypes</li> <li>Update ObjectTypes</li> <li>Read Instances</li> <li>Create instances</li> <li>Delete instances</li> <li>Update instances</li> <li>Read instance associations</li> <li>Create instance associations</li> <li>Delete instance associations</li> <li>Read ObjectTypeAssociations</li> <li>Create ObjectTypeAssociations</li> <li>Delete ObjectTypeAssociations</li> </ul> <div> <i>Note</i>  The scopes mentioned here for Thing Engineer are specific to Master Data. </div>	<cdconf>.r <cdconf>.c <cdconf>.d <cdconf>.u <cdinstance>.r <cdinstance>.c <cdinstance>.d <cdinstance>.u <thing>.r <thing>.c <thing>.d <cdconf>.r <cdconf>.c <cdconf>.d

## 9.6 Value Data Types for Object Types and Properties

Display the list of supported data types for ObjectType creation

Value Data Type	Data Type	Length Mandatory	Value
String	String	No	<div> <div>i Note</div> <p>Maximum permissible length is 255 characters.</p> <p>You can increase the length for a property type even if the data already exists for this property. However, you cannot decrease the length for a property regardless of whether the data exists or not.</p> <p>Special characters forward slash (/) and caret (^) are not supported.</p> </div> <div> <div>i Note</div> <p>Default length is 127 characters.</p> </div>
Numeric	Integer	No	<div> <div>i Note</div> <p>Maximum permissible value is 2,147,483,647.</p> <p>Minimum permissible value is -2,147,483,648</p> </div>

Value Data Type	Data Type	Length Mandatory	Value
Numeric	Decimal	Yes	
	<div> <div>i Note</div> <p>When you set a decimal value for a property type, you must indicate the precision upto which you require the value. This value is defined in the format (x,y), where:</p> <ul style="list-style-type: none"> <li>• X indicates the overall digits in the value. The permissible range is between 1 to 10 digits.</li> <li>• Y indicates the number of decimal or fractional digits in the value. The permissible range is between 1 to X.</li> </ul> <p>For example, for a value 32.45, the format defined is (4,2).</p> <p>For a value 12345.123, the format defined is (8,3).</p> </div>		
NumericFlexible	Float	No	Value range is -3.4E+38 to +3.4E+38
	<div> <div>i Note</div> <p>This is used to store 32 bit floating point values.</p> </div>		
ThingID	String	No	Default is 32 characters
	This is used to store 32 character ThingID.		
BPID	String	No	Default is 32 characters
Timestamp	Timestamp	No	Value range is 0001-01-01 00:00:00 to 9999-12-31 23:59:59.9999999
	Corresponds to UTC date and time (with a precision of 0.1 microseconds).		
DateTime	Date time	No	Values range is 0001-01-01 00:00:00 to 9999-12-31 23:59:59
	Corresponds to UTC date and time (with seconds precision)		
Date	Date	No	Value range is 0001-01-01 to 9999-12-31



Value Data Type	Data Type	Length Mandatory	Value
Boolean	Boolean For Boolean data type, length is not required. So, length must NOT be passed in the payload.	No	Values supported are <b>true</b> and <b>false</b> .

**i Note**

You cannot change a Boolean data type to another data type and vice versa.

## 9.7 Limitations

- Object Type Annotation supports the following two values. Other values are not accepted.
  - `com.sap.apptot.security:spi`
  - `com.sap.apptot.security:pii`
- Property annotations accept only the values that are available in the same package. Other random values are not accepted.
- `$expand` is supported only for Get Object Types by ID (Single Object Type) and not for Get all Object Types.
- Only the below `$expand` values are supported for Get Object Types:
  - Single Expand
    - `$expand = Properties`
    - `$expand = Descriptions`
    - `$expand = Annotations`
    - `$expand = Properties/Annotations`
    - `$expand = Properties`
  - Multiple Expand
    - `$expand = Descriptions, Properties/Annotations, Properties/Descriptions`
    - `$expand = Descriptions, Annotations`
    - `$expand = Properties, Annotations`
- Batch requests with `Content-ID` header are not supported.
- `$batch` is only supported for object type instances and instance associations.

## 9.8 Audit Logs

The audit log service provides means for applications and services to store the audit log records and to comply with the SAP product standards SEC-215 (security event log), SEC-254 (log read access to sensitive personal

data), SEC-265 (log changes to personal data), and SEC-257 (configuration change log). This is available for the following platforms:

- SAP HANA XS Advanced
- SAP CP Cloud Foundry @ AWS
- SAP CP Cloud Foundry @ OpenStack
- SAP CP Cloud Foundry @ Azure
- SAP CP Neo

## Writing Audit Logs for Applications

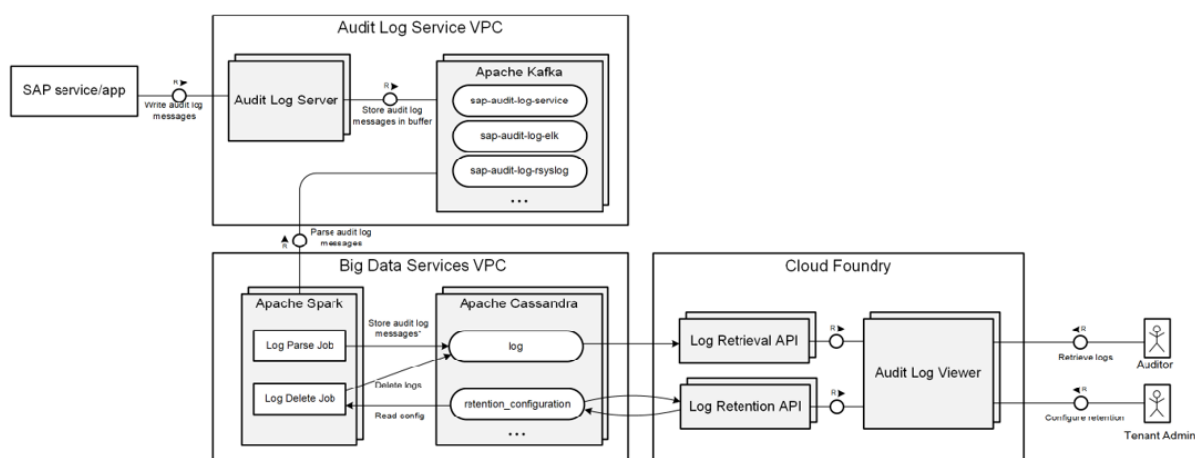
Cloud Foundry Applications should use the Audit Log Backing Service that is registered in the Cloud Foundry. The backing service is named `auditlog` and has only one unlimited standard plan.

## Unable to View Audit Log Service in an Organization and Space

The audit log service is enabled by default for a limited number of organizations and spaces. On AWS Canary, there is no such limitation and the service is enabled for all the organizations and spaces. If you do not see it in the Cloud Foundry marketplace (cf m), then it is not enabled for your organization. It is planned to have a self-service UI for requesting services to be enabled for your organization.

## Architecture

Following is the architecture of the data flow in audit logs:



\* Audit log messages from all Kafka topics (represent different sources of audit log messages) are parsed. Only the audit log messages from the Audit log Write REST are stored separately and log cleaned accordingly so that to be retrieved from the Retrieval API afterwards. They are separated currently based on subaccount or org and space if tenant does not exist. The Retention on subaccount level as well as the retention on org and space (will be removed in the future) are applied only on the audit logs written via the Auditlog Write REST API that are arranged by subaccount.

## Viewing Audit Logs

### SAP CP Cloud Foundry

A self-service audit log viewer is available for all the audit logs written via the Audit Log Service REST API.

#### Viewing of Audit Logs

Audit logs can only use the SaaS App `Auditlog Viewer`. An important implication due to the use of `Auditlog Viewer` is that audit logs written for the PaaS tenant until now with `.setTenant(null)` are no longer retrievable. They have to be logged for the tenant (Subaccount ID) of the PaaS app. Thus the `Auditlog Viewer` can only retrieve the audit logs created for the Subaccount, where it is subscribed.

#### Getting Access to `Auditlog Viewer`

Audit logs can be viewed using the SaaS App `Auditlog Viewer`. It is available for any Subaccount in the [Subscriptions](#) tab. To get access to it, ensure that you are a Subaccount Admin and subscribe to it.

#### Using `Auditlog Viewer`

Once logged in, the audit logs are retrieved by pressing the button placed on the right side of the time filter.

##### i Note

- All the logs that are logged for the respective Subaccount are shown.
- Organization, Space and AppID are not available.
- Searching by [To Date](#) is rounded to the next full hour.

### Audit log customer documentation

Audit Log Retrieval API on CF: <https://help.sap.com/viewer/65de2977205c403bbc107264b8eccf4b/Cloud/en-US/30ece35bac024ca69de8b16bff79c413.html>

Audit Log Viewer on CF: <https://help.sap.com/viewer/65de2977205c403bbc107264b8eccf4b/Cloud/en-US/e3baa5f1a0c64c44aac8ab3ea3d1b500.html>

### SAP CP Neo

A self-service retrieval API is available. See the documentation on how to use the Audit Log Retrieval API:

<https://help.sap.com/viewer/65de2977205c403bbc107264b8eccf4b/Cloud/en-US/e4d818da43af43e1983df8e9e5caadb2.html>

### SAP HANA XS Advanced

For SAP HANA XS Advanced with SAP Hana2 SP3, the audit logs are already stored in the Hana audit logging. For the SAP Hana installations prior Hana2 SP3, an audit log UI is still available and it is integrated into the XS Advanced Admin Tools. In order to use it, you can use the `XSA_ADMIN` user, which has the needed scopes

assigned. If you want to access it with a different user, you have to create a role collection and add the application role AuditLogViewer. The AuditLogViewer application role is a part of the application auditlog-ui and the role template AuditLogViewer. After that, you have to assign the role collection to the user.

## i Note

The Audit Log UI integrated in the XS Advanced Admin Tools is deprecated and expected to be removed with the upcoming versions for new installations.

## Viewing Audit Logs for Master Data

The Audit Log entries written by the Master Data application API is viewed as below.

Create, Update and Delete of an instance writes a Data Modification Message whereas Read of an instance writes a Data Access Message. Based on the requirement, filter can be set on the time, user, data and so on to check the corresponding Audit Logs.

The screenshot displays the AuditLog Viewer 1.0 interface. The top bar shows the user 'testuser' and two time filters: 'Feb 18, 2019, 9:48:55 AM' and 'Feb 18, 2019, 3:51:55 PM'. Below the bar is a table with columns: User, Time, Message, and Category. A single log entry is visible for 'testuser' at '18 Feb 2019, 15:51:28.023' with the category 'audit.data-modification'.

Below the table, the 'Log Entry Details' section provides a breakdown of the log entry:

- Timestamp:** 18 Feb 2019, 15:51:28.023 +0530
- Log Message:** Data Modification message. Attribute with name "Partner\_Name" was changed from "" to "Acme AG". Attribute with name "ObjectID" was changed from "" to "Partner-01". The attributes are a part of an object with type "ObjectType" and id consisting of: OT1 "Partner-01". They belong to a subject with type "Partner", role "Owns" and id consisting of: OBJECTID1 "sap.apptot.cd.g:noah\_citest01\_cdatademo\_Partner".
- IP:**
- User:** testuser
- Category:** audit.data-modification
- Data:** A JSON object containing details about the message, user, time, tenant, org, space, app, and data subject.

```

1 - {
2   "message_uuid": "1a7005b1-dc25-4d7c-96bd-5ca2b09f3590",
3   "time": "2019-02-18T10:21:28.023Z",
4   "tenant": "6b37714a-743c-49a9-bbfa-7245c0716662",
5   "org_id": "92f1da92-e5b3-4cc5-8c90-964165af11c8",
6   "space_id": "92f1da92-e5b3-4cc5-8c90-964165af11c8",
7   "app_or_service_id": "92f1da92-e5b3-4cc5-8c90-964165af11c8",
8   "als_service_id": "fa7d5d39-2ef9-456c-8ccc-2d3044a9d3b9",
9   "user": "testuser",
10  "category": "audit.data-modification",
11  "format_version": "",
12  "message": {
13    "uid": "1a7005b1-dc25-4d7c-96bd-5ca2b09f3590",
14    "user": "testuser",
15    "time": "2019-02-18T10:21:28.023Z",
16    "object": {
17      "type": "ObjectType",
18      "id": {
19        "OT1": "Partner-01"
20      }
21    },
22    "data_subject": {
23      "type": "Partner",
24      "role": "Owns",
25      "id": {

```

### Data Modification Message

Timestamp	18 Feb 2019, 10:52:03.407 +0530
Log Message	<p>Data Access message.</p> <p>Attribute with name "CreatedByUser" was read.</p> <p>Attribute with name "LastChangedByUser" was read.</p> <p>The attributes are a part of an object with type "ThingType" and id consisting of: THINGTYPE "noah.citest01.qk0g3.core.refproptest:Car". They belong to a subject with type "Thing", role "Owns" and id consisting of: THINGID "014C4535C33AA4557AE3F7AA42DD2C918". The message has the following attachments: .</p>
IP	
User	testuser
Category	audit.data-access
Data	<pre> 1- { 2-   "message_uuid": "Zeda05eb-3b2d-4234-8772-95aa81864193", 3-   "time": "2019-02-18T05:22:03.407Z", 4-   "tenant": "6b37714a-743c-49a9-bbfa-7245c0716662", 5-   "org_id": "92f1da92-e5b3-4cc5-8c90-964165af11c8", 6-   "space_id": "92f1da92-e5b3-4cc5-8c90-964165af11c8", 7-   "app_or_service_id": "92f1da92-e5b3-4cc5-8c90-964165af11c8", 8-   "als_service_id": "fa7d5d39-2ef9-456c-8ccc-2d3044a9d3b9", 9-   "user": "testuser", 10-  "category": "audit.data-access", 11-  "format_version": "", 12-  "message": { 13-    "uuid": "Zeda05eb-3b2d-4234-8772-95aa81864193", 14-    "user": "testuser", 15-    "time": "2019-02-18T05:22:03.407Z", 16-    "channel": "SAP IoT AE Services GET_THINGS", 17-    "object": { 18-      "type": "ThingType", 19-      "id": { 20-        "THINGTYPE": "noah.citest01.qk0g3.core.refproptest:Car" 21-      } 22-    }, 23-    "data_subject": { 24-      "type": "Thing", 25-      "role": "Owns", </pre>

## 9.9 Data Protection and Privacy

You can assign the predefined annotations for object types during the following action:

- After you define the annotations, all properties of the object type are considered as personal data or sensitive data based on the annotation defined. You can remove or add the annotation for the object type, only if data does not exist for the properties of the object type. Whenever you create or modify data for properties that are identified as personal data or sensitive data, the system must generate audit logs.

## Object Type Configuration (OData Service)

In the following example payload for creating an object type, the predefined annotation `com.sap.appiot.security:spi` is assigned to the object type `iotae.devtest1.cdemo:Partner` to indicate that all properties of the object type are considered as sensitive data:

Request payload to create an object type with sensitive personal annotation:

`/CustomDataConfiguration/v1/Packages('iotae.devtest1.cdemo')/ObjectTypes`

```
{
  "Name": "iotae.devtest1.cdemo:Partner",
  "PackageName": "iotae.devtest1.cdemo",
  "Descriptions": [{
    "LanguageCode": "en",
    "Description" : "Partner Data"
  }],
  "Source": "CRM_Data",
  "Annotations": [
    {
      "Name": "com.sap.appiot.security:spi"
    }
  ],
  "Properties": [
    {
      "Name": "Partner_Name",
      "Descriptions": [
        {
          "LanguageCode": "en",
          "Description" : "Partner Name"
        }
      ],
      "Type": "String",
      "PropertyLength": "120"
    }
  ]
}
```

# 10 Custom Master Data Composite Services

In the previous section, we have seen the use of [Custom Master Data \[page 598\]](#) APIs. In order to obtain information about custom master data instances which are associated with a thing (using the custom master data APIs), a user must first use [Read an Instance Association \[page 649\]](#) to fetch the master data instance associated with the thing, and then use [Read an Object Type Instance \[page 637\]](#) to get details about the custom master data instance. Custom Master Data Composite Services allow the users to retrieve the same information without having to make multiple calls, that is, they can just use one composite API to get the data.

The Custom Master Data Composite Services consist of two services: [Composite Custom Master Data Basic Service \[page 667\]](#) and [Composite Custom Master Data Details Service \[page 675\]](#). The basic service retrieves a list of things along with the basic details of the custom master data. The details service retrieves a list of things along with their custom master data details, based on the object types specified in the request header. The service retrieves metadata and data for object types that are common across specified thing types.

## i Note

Users are advised not to use underscores in ObjectType names. Names with underscores will give empty results. For instance, `iotae.devtest1.cdemo:Partner_Info` will give empty results for ObjectType Instances.

## 10.1 Composite Custom Master Data Basic Service

The service retrieves a list of things along with the basic details of the custom master data.

OData Version: 2.0

**Root URI:** `http://<server address>[:<port number>]/CompositeMasterData/v1/`

Supported OData Features

Feature	Support
Query options	<p>The current implementation supports the following query options, which can be passed as query or path parameters:</p> <ul style="list-style-type: none"><li>• \$skip</li><li>• \$top</li><li>• \$orderby</li><li>• \$filter</li><li>• \$select</li><li>• \$expand</li><li>• \$inlinecount</li></ul>

## 10.1.1 Read Metadata of Composite Custom Master Data Basic Service

This service retrieves the metadata of Composite Custom Master Data Basic Service.

### Request

**URI:** /CompositeMasterData/v1/\$metadata

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <compcd>.r

### Request Parameters

None

### Request Example

```
/CompositeMasterData/v1/$metadata
```

### Response

#### Response Status and Error Codes

Code	Description
200	Metadata retrieved successfully

### Payload

Only media type supported is **XML**. The **XML** for this method has the structure as shown in the following example.

### Example

```
/CompositeMasterData/v1/$metadata
```

```
<?xml version="1.0" ?>
<edmx:Edmx Version="1.0" xmlns:edmx="http://schemas.microsoft.com/ado/2007/06/edmx">
  <edmx:DataServices m:DataServiceVersion="1.0" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:sap="http://www.sap.com/Protocols/SAPData">
    <Schema Namespace="com.sap.apptot" xmlns="http://schemas.microsoft.com/ado/2008/09/edm">
      <EntityType Name="ObjectInstance">
        <Key>
          <PropertyRef Name="ObjectType"></PropertyRef>
```



```

        <PropertyRef Name="ObjectId"></PropertyRef>
    </Key>
    <Property Name="ObjectId" Type="Edm.String"></Property>
    <Property Name="ObjectType" Type="Edm.String"></Property>
    <Property Name="StartTime" Type="Edm.DateTime"></Property>
    <Property Name="ExpiryTime" Type="Edm.DateTime"></Property>
</EntityType>
<EntityType Name="Thing">
    <Key>
        <PropertyRef Name="ThingId"></PropertyRef>
    </Key>
    <Property Name="ThingId" Type="Edm.String" Nullable="false">
        <Documentation>
            <LongDescription>Thing unique ID</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingDescription" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing description</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingName" Type="Edm.String" Nullable="false">
        <Documentation>
            <LongDescription>Thing name</LongDescription>
        </Documentation>
    </Property>
    <Property Name="AlternateId" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing AlternateId</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingExternalId" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing external id</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingType" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing Type ID associated with the
Thing</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ObjectGroup" Type="Edm.String">
        <Documentation>
            <LongDescription>Object group used to create the Thing</
LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCustomerId" Type="Edm.String">
        <Documentation>
            <LongDescription>Business partner ID associated for the
Thing</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingLocationId" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing location ID</LongDescription>
        </Documentation>
    </Property>
    <NavigationProperty Name="ObjectInstance"
Relationship="com.sap.apptot.ObjectInstance" FromRole="ThingEntities"
ToRole="ObjectInstanceEntities"></NavigationProperty>
</EntityType>
    <Association Name="ObjectInstance">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.ObjectInstance" Multiplicity="*"
Role="ObjectInstanceEntities"></End>

```

```

        </Association>
        <EntityContainer Name="CompositeMasterDataAnalytics"
m:IsDefaultEntityContainer="true">
            <EntitySet Name="ObjectInstances"
EntityType="com.sap.appiot.ObjectInstance"></EntitySet>
            <EntitySet Name="Things" EntityType="com.sap.appiot.Thing"></
EntitySet>
            <AssociationSet Name="ObjectInstance"
Association="com.sap.appiot.ObjectInstance">
                <End EntitySet="Things" Role="ThingEntities"></End>
                <End EntitySet="ObjectInstances"
Role="ObjectInstanceEntities"></End>
            </AssociationSet>
        </EntityContainer>
    </Schema>
</edmx:DataServices>
</edmx:Edmx>

```

## 10.1.2 Read List of Things

This service retrieves a list of things along with custom master data.

### i Note

The service supports \$expand query parameter in the request URL with which you can see the details of the custom master data in your result set.

By default, the service displays a maximum of 100 records per response. You can navigate to the next set of results, if available, using the \_\_next link at the end in the response payload. However, the number of records retrieved depends on the query parameter \$top.

## Request

**URI:** /CompositeMasterData/v1/Things

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <compcd>.r

### Request Headers

None

### Request Parameters

None

## Query Request Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set
\$skip	No	Integer	Number of the first n records to be excluded from the result set
\$inlinecount	No	String	<p>Total number of things available; permissible values are <i>allpages</i> and <i>none</i>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"><li>• If you use this with the \$top and \$skip parameters, the result contains the total count of things available.</li><li>• If you use this with \$filter parameter, the result contains the total count of things based on the defined filter condition.</li></ul>
\$count	No	Boolean	<p>Total number of things available based on the current user's authorizations; permissible values are true and false.</p> <p>If this is used in conjunction with the \$top and \$skip parameters, the result contains the total count of things available.</p>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; all thing entity fields are valid.</p> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>
\$select		String	Select condition for the thing entity fields to be included in the result set.
\$filter		String	<p>Filter condition to be applied; all thing entity fields and fields in navigation properties of object types are valid.</p> <p>Multiple filters are supported using the <i>and</i> separator.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using \$filter.</p>

## Request Examples

```
/CompositeMasterData/v1/Things
```

Assume, there are 1000 records in the database. Due to performance reasons, the method retrieves the first 100 records and displays a next link to navigate to the next set of records.

```
/CompositeMasterData/v1/Things?$top=500
```

Assume, there are 1000 records in the database. Due to performance reasons, the method retrieves the first 100 records and displays a next link to navigate to the next set of records until the top 500 records are retrieved.

## Response

### Response Status and Error Codes

Code	Description
200	Thing list retrieved successfully

### Payload

Format: [JSON](#)

Media types supported are **JSON** and **XML**. The **JSON** for this method has the structure as depicted in the following example.

### Example

The result set contains all things along with the basic details of custom master data.

**Request URL Path:** /CompositeMasterData/v1/Things?\$expand=ObjectInstance

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/v1/Things('1A0A556907E44E64AF9D983B116105BB')",
          "uri": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/v1/Things('1A0A556907E44E64AF9D983B116105BB')",
          "type": "com.sap.apptot.Thing"
        },
        "ThingId": "1A0A556907E44E64AF9D983B116105BB",
        "ThingDescription": "Cooler001 Description",
        "ThingName": "Cooler001",
        "AlternateId": null,
        "ThingExternalId": "Cooler001",
        "ThingType": "core.cooler.package:CoolerType",
        "ObjectGroup": "693FE2A1823E45149ECDA575CC64C006",
        "ThingCustomerId": null,
        "ThingLocationId": null,
        "ObjectInstance": {
          "results": [
            {
              "ObjectId": "D062089C1F8B4E8AA1803455F701BF20",
              "ObjectType": "core.cooler.package:Product",
              "StartTime": "/Date(1558616456000)/",
              "ExpiryTime": "/Date(1560686456000)/"
            },
            {
              "ObjectId": "63BF2770BFF24FE295A969C25FEE4402",
              "ObjectType": "core.cooler.package:Address",
              "StartTime": "/Date(1558616456000)/",
              "ExpiryTime": "/Date(1561118456000)/"
            }
          ]
        }
      }
    ]
  }
}
```

```

    ]
  },
  {
    "__metadata": {
      "id": "https://composite-custom-master-data-
sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/v1/
Things('7A4F53C0C38F46E0874A66A3635BEB64')",
      "uri": "https://composite-custom-master-data-
sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/v1/
Things('7A4F53C0C38F46E0874A66A3635BEB64')",
      "type": "com.sap.apptot.Thing"
    },
    "ThingId": "7A4F53C0C38F46E0874A66A3635BEB64",
    "ThingDescription": "Cooler006 Description",
    "ThingName": "Cooler006",
    "AlternateId": null,
    "ThingExternalId": "Cooler006",
    "ThingType": "core.cooler.package:CoolerType",
    "ObjectGroup": "693FE2A1823E45149ECDA575CC64C006",
    "ThingCustomerId": null,
    "ThingLocationId": null,
    "ObjectInstance": {
      "results": [
        {
          "ObjectId": "B5EE38ADCC8E4E8BAD105848AC682D0A",
          "ObjectType": "core.cooler.package:Product",
          "StartTime": "/Date(1558616456000)/",
          "ExpiryTime": "/Date(1560945656000)/"
        },
        {
          "ObjectId": "7B39B8589E5A4915B387086A89AAB1CF",
          "ObjectType": "core.cooler.package:Address",
          "StartTime": "/Date(1558616456000)/",
          "ExpiryTime": "/Date(1561896056000)/"
        }
      ]
    }
  },
  {
    "__metadata": {
      "id": "https://composite-custom-master-data-
sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/v1/
Things('9208196DC0A4445B84D37D66AB614C8D')",
      "uri": "https://composite-custom-master-data-
sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/v1/
Things('9208196DC0A4445B84D37D66AB614C8D')",
      "type": "com.sap.apptot.Thing"
    },
    "ThingId": "9208196DC0A4445B84D37D66AB614C8D",
    "ThingDescription": "Cooler001 Description",
    "ThingName": "Cooler002",
    "AlternateId": null,
    "ThingExternalId": "Cooler002",
    "ThingType": "core.cooler.package:CoolerType",
    "ObjectGroup": "693FE2A1823E45149ECDA575CC64C006",
    "ThingCustomerId": null,
    "ThingLocationId": null,
    "ObjectInstance": {
      "results": [
        {
          "ObjectId": "A365DE2FBAEB4BF18B0B30C70E1B925D",
          "ObjectType": "core.cooler.package:Product",
          "StartTime": "/Date(1558616456000)/",
          "ExpiryTime": "/Date(1563537656000)/"
        },
        {
          "ObjectId": "051C3968885B408680D55A4A3F5FE362",

```

```

        "ObjectType": "core.cooler.package:Address",
        "StartTime": "/Date(1558616456000)/",
        "ExpiryTime": "/Date(1564488056000)/"
    }
}
},
{
    "__metadata": {
        "id": "https://composite-custom-master-data-
sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/v1/
Things('B3407370CFCF4F4EADA1BDCD7472E3EE')",
        "uri": "https://composite-custom-master-data-
sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/v1/
Things('B3407370CFCF4F4EADA1BDCD7472E3EE')",
        "type": "com.sap.apptot.Thing"
    },
    "ThingId": "B3407370CFCF4F4EADA1BDCD7472E3EE",
    "ThingDescription": "Cooler005 Description",
    "ThingName": "Cooler005",
    "AlternateId": null,
    "ThingExternalId": "Cooler005",
    "ThingType": "core.cooler.package:CoolerType",
    "ObjectGroup": "693FE2A1823E45149ECDA575CC64C006",
    "ThingCustomerId": null,
    "ThingLocationId": null,
    "ObjectInstance": {
        "results": [
            {
                "ObjectId": "8E70E64462214A93A2169B2C89136573",
                "ObjectType": "core.cooler.package:Product",
                "StartTime": "/Date(1558616456000)/",
                "ExpiryTime": "/Date(1562760056000)/"
            },
            {
                "ObjectId": "4522AB63B79547AB9CCE005E6287F4C6",
                "ObjectType": "core.cooler.package:Address",
                "StartTime": "/Date(1558616456000)/",
                "ExpiryTime": "/Date(1561896056000)/"
            }
        ]
    }
}
},
{
    "__metadata": {
        "id": "https://composite-custom-master-data-
sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/v1/
Things('DAAF7B9BCDC6421C95629CA984627628')",
        "uri": "https://composite-custom-master-data-
sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/v1/
Things('DAAF7B9BCDC6421C95629CA984627628')",
        "type": "com.sap.apptot.Thing"
    },
    "ThingId": "DAAF7B9BCDC6421C95629CA984627628",
    "ThingDescription": "Cooler003 Description",
    "ThingName": "Cooler003",
    "AlternateId": null,
    "ThingExternalId": "Cooler003",
    "ThingType": "core.cooler.package:CoolerType",
    "ObjectGroup": "693FE2A1823E45149ECDA575CC64C006",
    "ThingCustomerId": null,
    "ThingLocationId": null,
    "ObjectInstance": {
        "results": [
            {
                "ObjectId": "FD8EE8B1A322462689C4E4ACC2CEB3DB",
                "ObjectType": "core.cooler.package:Product",
                "StartTime": "/Date(1558616456000)/",

```

```

    },
    {
      "ObjectId": "24117A3D559B4153A0183A50A818B124",
      "ObjectType": "core.cooler.package:Address",
      "StartTime": "/Date(1558616456000)/",
      "ExpiryTime": "/Date(1561377656000)/"
    }
  ]
}

```

## 10.2 Composite Custom Master Data Details Service

This service retrieves a list of things along with their custom master data details, based on the object types specified in the request header. The service retrieves metadata and data for object types that are common across specified thing types.

OData Version: 2.0

**Root URI:** http://<server address>[:<port number>]/CompositeMasterData/Details/v1/

## Supported OData Features

Feature	Support
Query options	<p>The current implementation supports the following query options, which can be passed as query or path parameters:</p> <ul style="list-style-type: none"><li>• \$skip</li><li>• \$top</li><li>• \$orderby</li><li>• \$filter</li><li>• \$select</li><li>• \$expand</li><li>• \$inlinecount</li></ul>

### 10.2.1 Read Metadata of Composite Custom Master Data Details Service

This service retrieves the metadata for object types that are common across thing types. It retrieves the metadata of Composite Custom Master Data Details Service.

## i Note

If you do not specify the header parameters, the service retrieves only the thing entity.

## Request

**URI:** /CompositeMasterData/Details/v1/\$metadata

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <compcd>.r

### Request Headers

Header	Required	Values
sap-iot-ot	No	Object types separated by comma ","

### Request Parameters

None

### Request Example

```
/CompositeMasterData/Details/v1/$metadata
```

## Response

### Response Status and Error Codes

Code	Description
200	Metadata retrieved successfully

### Payload

Only media type supported is **XML**. The **XML** for this method has the structure as shown in the following example.

### Example

```
/CompositeMasterData/Details/v1/$metadata
```

Request header parameters sap-iot-ot with value core.cooler.package:Product,core.cooler.package:Address

```
<?xml version="1.0" ?>
<edmx:Edmx Version="1.0" xmlns:edmx="http://schemas.microsoft.com/ado/2007/06/edmx">
  <edmx:DataServices m:DataServiceVersion="1.0" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:sap="http://www.sap.com/Protocols/SAPData">
    <Schema Namespace="com.sap.appiot" xmlns="http://schemas.microsoft.com/ado/2008/09/edm">
      <EntityType Name="Thing">
        <Key>
          <PropertyRef Name="ThingId"></PropertyRef>
        </Key>
      </EntityType>
    </Schema>
  </DataServices>
</edmx:Edmx>
```



```

        <Property Name="ThingId" Type="Edm.String" Nullable="false">
            <Documentation>
                <LongDescription>Thing unique ID</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingDescription" Type="Edm.String">
            <Documentation>
                <LongDescription>Thing description</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingName" Type="Edm.String" Nullable="false">
            <Documentation>
                <LongDescription>Thing name</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingExternalId" Type="Edm.String">
            <Documentation>
                <LongDescription>Thing external id</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingType" Type="Edm.String">
            <Documentation>
                <LongDescription>Thing Type ID associated with the
Thing</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ObjectGroup" Type="Edm.String">
            <Documentation>
                <LongDescription>Object group used to create the Thing</
LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingCustomerId" Type="Edm.String">
            <Documentation>
                <LongDescription>Business partner ID associated for the
Thing</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingLocationId" Type="Edm.String">
            <Documentation>
                <LongDescription>Thing location ID</LongDescription>
            </Documentation>
        </Property>
        <NavigationProperty Name="DYN_ENT_core_cooler__Product"
Relationship="com.sap.apptot.DYN_ENT_core_cooler__Product"
FromRole="ThingEntities" ToRole="DYN_ENT_core_cooler__ProductEntities"></
NavigationProperty>
        <NavigationProperty Name="DYN_ENT_core_cooler__Address"
Relationship="com.sap.apptot.DYN_ENT_core_cooler__Address"
FromRole="ThingEntities" ToRole="DYN_ENT_core_cooler__AddressEntities"></
NavigationProperty>
    </EntityType>
    <EntityType Name="DYN_ENT_core_cooler__Product">
        <Key>
            <PropertyRef Name="ObjectType"></PropertyRef>
            <PropertyRef Name="ObjectID"></PropertyRef>
        </Key>
        <Property Name="ObjectType" Type="Edm.String"></Property>
        <Property Name="ObjectID" Type="Edm.String"></Property>
        <Property Name="core.cooler.package.Product.ProductName"
Type="Edm.String"></Property>
        <Property Name="core.cooler.package.Product.ProductDescription"
Type="Edm.String"></Property>
        <Property Name="core.cooler.package.Product.StartTime"
Type="Edm.DateTime"></Property>
        <Property Name="core.cooler.package.Product.ExpiryTime"
Type="Edm.DateTime"></Property>
    </EntityType>

```

```

        <EntityType Name="DYN_ENT_core_cooler__Address">
            <Key>
                <PropertyRef Name="ObjectType"></PropertyRef>
                <PropertyRef Name="ObjectID"></PropertyRef>
            </Key>
            <Property Name="ObjectType" Type="Edm.String"></Property>
            <Property Name="ObjectID" Type="Edm.String"></Property>
            <Property Name="core.cooler.package.Address.Country"
Type="Edm.String"></Property>
            <Property Name="core.cooler.package.Address.City"
Type="Edm.String"></Property>
            <Property Name="core.cooler.package.Address.StartTime"
Type="Edm.DateTime"></Property>
            <Property Name="core.cooler.package.Address.ExpiryTime"
Type="Edm.DateTime"></Property>
        </EntityType>
        <Association Name="DYN_ENT_core_cooler__Product">
            <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
            <End Type="com.sap.apptot.DYN_ENT_core_cooler__Product"
Multiplicity="*" Role="DYN_ENT_core_cooler__ProductEntities"></End>
        </Association>
        <Association Name="DYN_ENT_core_cooler__Address">
            <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
            <End Type="com.sap.apptot.DYN_ENT_core_cooler__Address"
Multiplicity="*" Role="DYN_ENT_core_cooler__AddressEntities"></End>
        </Association>
        <EntityContainer Name="CompositeMasterDataAnalytics"
m:IsDefaultEntityContainer="true">
            <EntitySet Name="Things" EntityType="com.sap.apptot.Thing"></
EntitySet>
            <EntitySet Name="DYN_ENT_core_cooler__Products"
EntityType="com.sap.apptot.DYN_ENT_core_cooler__Product"></EntitySet>
            <EntitySet Name="DYN_ENT_core_cooler__Addressss"
EntityType="com.sap.apptot.DYN_ENT_core_cooler__Address"></EntitySet>
            <AssociationSet Name="DYN_ENT_core_cooler__Product"
Association="com.sap.apptot.DYN_ENT_core_cooler__Product">
                <End EntitySet="Things" Role="ThingEntities"></End>
                <End EntitySet="DYN_ENT_core_cooler__Products"
Role="DYN_ENT_core_cooler__ProductEntities"></End>
            </AssociationSet>
            <AssociationSet Name="DYN_ENT_core_cooler__Address"
Association="com.sap.apptot.DYN_ENT_core_cooler__Address">
                <End EntitySet="Things" Role="ThingEntities"></End>
                <End EntitySet="DYN_ENT_core_cooler__Addressss"
Role="DYN_ENT_core_cooler__AddressEntities"></End>
            </AssociationSet>
        </EntityContainer>
    </Schema>
</edmx:DataServices>
</edmx:Edmx>

```

## 10.2.2 Read List of Things

This service retrieves a list of things for the specified object types of custom master data.

With this service, you can retrieve the list of things based on the following request header parameter:

**Object type** – You can list one or more object types separated by comma.

Service retrieves all things of thing type that are associated with the specified object types. The request header parameters are optional.

## Note

The service supports \$expand query parameter in the request URL with which you can see the details of the custom master data in your result set. However, when the values provided in the request header parameter and in the \$expand query parameter differ, the values specified in the \$expand query parameter takes the precedence and the result set is retrieved only based on them. The request header parameters are ignored in such cases.

By default, the service displays a maximum of 100 records per response. You can navigate to the next set of results, if available, using the \_\_next link at the end in the response payload. However, the number of records retrieved depends on the query parameter \$top.

The table below illustrates the available things, thing types, and object types.

Things	Thing Type	Object Type
T1, T2	TT1	OT1, OT2
T3, T4	TT2	OT2, OT3, OT4
T5	TT3	OT1, OT2, OT4

The table below illustrates how data is retrieved for a given combination of request header parameters and \$expand query parameters:

Request	Request Header Parameters	Result Set Contains...	Based on the...
/things	OT1, OT2	T1, T2	Request header parameters
/things?\$expand=OT2	OT1, OT2	T1, T2, T3, T4, T5	\$expand query parameters (request header parameters are ignored)
/things?\$expand=OT1, OT2	OT1, OT2	T1, T2	Request header parameters and \$expand query parameters are same

## Request

**URI:** /CompositeMasterData/Details/v1/Things

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <compcd>.r

## Request Headers

Header	Required	Values
sap-iot-ot	No	Object types separated by comma ","

## Request Parameters

None

## Query Request Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set
\$skip	No	Integer	Number of the first n records to be excluded from the result set
\$inlinecount	No	String	<p>Total number of things available; permissible values are <a href="#">allpages</a> and <a href="#">none</a>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"><li>• If you use this with the \$top and \$skip parameters, the result contains the total count of things available.</li><li>• If you use this with \$filter parameter, the result contains the total count of things based on the defined filter condition.</li></ul>
\$count	No	Boolean	<p>Total number of things available based on the current user's authorizations; permissible values are true and false.</p> <p>If this is used in conjunction with the \$top and \$skip parameters, the result contains the total count of things available.</p>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; all thing entity fields are valid.</p> <p>To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</p>
\$select		String	Select condition for the thing entity fields to be included in the result set.
\$filter		String	<p>Filter condition to be applied; all thing entity fields and fields in navigation properties of object types are valid.</p> <p>Multiple filters are supported using the <a href="#">and</a> separator.</p> <p>You can use the string functions such as <a href="#">substringof</a>, <a href="#">startswith</a>, and <a href="#">endswith</a> to search data using \$filter.</p>

## Request Examples

```
/CompositeMasterData/Details/v1/Things
```

Assume there are 1000 records in the database. Due to performance reasons, the method retrieves the first 100 records and displays a next link to navigate to the next set of records.

```
/CompositeMasterData/v1/Things?$top=500
```

Assume there are 1000 records in the database. Due to performance reasons, the method retrieves the first 100 records and displays a next link to navigate to the next set of records until the top 500 records are retrieved.

## Response

### Response Status and Error Codes

Code	Description
200	Thing list retrieved successfully

### Payload

Format: [JSON](#)

Media types supported are **JSON** and **XML**. The **JSON** for this method has the structure as depicted in the following example.

### Example

**Request URL Path:** CompositeMasterData/Details/v1/Things?

\$expand=DYN\_ENT\_core\_cooler\_\_Product,DYN\_ENT\_core\_cooler\_\_Address

Request header parameter `sap-iot-ot` with value `core.cooler.package:Product,core.cooler.package:Address`

The result set contains all things that belong to the thing types associated with the object types specified in the request header.

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/Details/v1/Things('1A0A556907E44E64AF9D983B116105BB')",
          "uri": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/Details/v1/Things('1A0A556907E44E64AF9D983B116105BB')",
          "type": "com.sap.apptot.Thing"
        },
        "ThingId": "1A0A556907E44E64AF9D983B116105BB",
        "ThingDescription": "Cooler001 Description",
        "ThingName": "Cooler001",
        "AlternateId": null,
        "ThingExternalId": "Cooler001",
      }
    ]
  }
}
```

```

"core.cooler.package:CoolerType",
"693FE2A1823E45149ECDA575CC64C006",
"ThingCustomerId": null,
"ThingLocationId": null,
"DYN_ENT_core_cooler__Product": {
  "results": [
    {
      "ObjectType": "core.cooler.package:Product",
      "ObjectId": "D062089C1F8B4E8AA1803455F701BF20",
      "core.cooler.package.Product.ProductName":
"CocaCola",
      "core.cooler.package.Product.ProductDescription":
"CocaCola",
      "core.cooler.package.Product.StartTime": "/"
Date(1558616456000)/",
      "core.cooler.package.Product.ExpiryTime": "/"
Date(1560686456000)/"
    }
  ],
  "DYN_ENT_core_cooler__Address": {
    "results": [
      {
        "ObjectType": "core.cooler.package:Address",
        "ObjectId": "63BF2770BFF24FE295A969C25FEE4402",
        "core.cooler.package.Address.Country": "India",
        "core.cooler.package.Address.City": "Bangalore",
        "core.cooler.package.Address.StartTime": "/"
Date(1558616456000)/",
        "core.cooler.package.Address.ExpiryTime": "/"
Date(1561118456000)/"
      }
    ],
    {
      "__metadata": {
        "id": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:443/
CompositeMasterData/Details/v1/Things('7A4F53C0C38F46E0874A66A3635BEB64')",
        "uri": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:
443/CompositeMasterData/Details/v1/Things('7A4F53C0C38F46E0874A66A3635BEB64')",

```

```

"type": "com.sap.appiot.Thing"
},
"ThingId":
"7A4F53C0C38F46E0874A66A3635BEB64",
"ThingDescription": "Cooler006 Description",
"ThingName":
"Cooler006",
"AlternateId":
null,
"ThingExternalId": "Cooler006",
"ThingType":
"core.cooler.package:CoolerType",
"ObjectGroup":
"693FE2A1823E45149ECDA575CC64C006",
"ThingCustomerId": null,
"ThingLocationId": null,
"DYN_ENT_core_cooler__Product": {
"results": [
{
"ObjectType": "core.cooler.package:Product",
"ObjectId": "B5EE38ADCC8E4E8BAD105848AC682D0A",
"core.cooler.package.Product.ProductName":
"Pepsi",
"core.cooler.package.Product.ProductDescription":
"Pepsi",
"core.cooler.package.Product.StartTime": "/"
Date(1558616456000)/",
"core.cooler.package.Product.ExpiryTime": "/"
Date(1560945656000)/"
}
],
},
"DYN_ENT_core_cooler__Address": {
"results": [
{
"ObjectType": "core.cooler.package:Address",
"ObjectId": "7B39B8589E5A4915B387086A89AAB1CF",
"core.cooler.package.Address.Country": "India",
"core.cooler.package.Address.City": "Hyderabad",
"core.cooler.package.Address.StartTime": "/"
Date(1558616456000)/",
"core.cooler.package.Address.ExpiryTime": "/"
Date(1561896056000)/"
}
]
}

```

```

    }
    ],
    },
    {
        "__metadata": {
            "id": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/Details/v1/Things('9208196DC0A4445B84D37D66AB614C8D')",
            "uri": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/Details/v1/Things('9208196DC0A4445B84D37D66AB614C8D')",
            "type": "com.sap.apptot.Thing"
        },
        "ThingId":
        "9208196DC0A4445B84D37D66AB614C8D",
        "ThingDescription": "Cooler001 Description",
        "ThingName":
        "Cooler002",
        "AlternateId":
        null,
        "ThingExternalId": "Cooler002",
        "ThingType":
        "core.cooler.package:CoolerType",
        "ObjectGroup":
        "693FE2A1823E45149ECDA575CC64C006",
        "ThingCustomerId": null,
        "ThingLocationId": null,
        "DYN_ENT_core_cooler__Product": {
            "results": [
                {
                    "ObjectType": "core.cooler.package:Product",
                    "ObjectId": "A365DE2FBAEB4BF18B0B30C70E1B925D",
                    "core.cooler.package.Product.ProductName":
                    "Sprite",
                    "core.cooler.package.Product.ProductDescription":
                    "Sprite",
                    "core.cooler.package.Product.StartTime": "/"
                    Date(1558616456000)/",
                    "core.cooler.package.Product.ExpiryTime": "/"
                    Date(1563537656000)/"
                }
            ]
        },
        "DYN_ENT_core_cooler__Address": {
            "results": [
                {
                    "ObjectType": "core.cooler.package:Address",

```



```

        "ObjectId": "051C3968885B408680D55A4A3F5FE362",
        "core.cooler.package.Address.Country": "India",
        "core.cooler.package.Address.City": "Chennai",
        "core.cooler.package.Address.StartTime": "/"
Date(1558616456000) /",
        "core.cooler.package.Address.ExpiryTime": "/"
Date(1564488056000) /"
    }
}

},
{
    "__metadata": {
        "id": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/Details/v1/Things('B3407370CFCF4F4EADA1BD7472E3EE')",
        "uri": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:443/CompositeMasterData/Details/v1/Things('B3407370CFCF4F4EADA1BD7472E3EE')",
        "type": "com.sap.apptot.Thing"
    },
    "ThingId":
    "B3407370CFCF4F4EADA1BD7472E3EE",
    "ThingDescription": "Cooler005 Description",
    "ThingName":
    "Cooler005",
    "AlternateId":
    null,
    "ThingExternalId": "Cooler005",
    "ThingType":
    "core.cooler.package:CoolerType",
    "ObjectGroup":
    "693FE2A1823E45149ECDA575CC64C006",
    "ThingCustomerId": null,
    "ThingLocationId": null,
    "DYN_ENT_core_cooler__Product": {
        "results": [
            {
                "ObjectType": "core.cooler.package:Product",
                "ObjectId": "8E70E64462214A93A2169B2C89136573",
                "core.cooler.package.Product.ProductName":
                "ThumsUp",
                "core.cooler.package.Product.ProductDescription":
                "ThumsUp",
                "core.cooler.package.Product.StartTime": "/"
Date(1558616456000) /",
                "core.cooler.package.Product.ExpiryTime": "/"
Date(1562760056000) /"
            }
        ]
    }
}

```

```

    }
    ],
    "DYN_ENT_core_cooler__Address": {
    "results": [
        {
            "ObjectType": "core.cooler.package:Address",
            "ObjectId": "4522AB63B79547AB9CCE005E6287F4C6",
            "core.cooler.package.Address.Country": "India",
            "core.cooler.package.Address.City": "Mumbai",
            "core.cooler.package.Address.StartTime": "/"
Date(1558616456000)/",
            "core.cooler.package.Address.ExpiryTime": "/"
Date(1561896056000)/"
        }
    ]
    },
    {
        "__metadata": {
            "id": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:443/
CompositeMasterData/Details/v1/Things('DAAF7B9BCDC6421C95629CA984627628')",
            "uri": "https://composite-custom-master-data-sap.cfapps.sap.hana.ondemand.com:
443/CompositeMasterData/Details/v1/Things('DAAF7B9BCDC6421C95629CA984627628')",
            "type": "com.sap.apptot.Thing"
        },
        "ThingId":
"DAAF7B9BCDC6421C95629CA984627628",
        "ThingDescription": "Cooler003 Description",
        "ThingName":
"Cooler003",
        "AlternateId":
null,
        "ThingExternalId": "Cooler003",
        "ThingType":
"core.cooler.package:CoolerType",
        "ObjectGroup":
"693FE2A1823E45149ECDA575CC64C006",
        "ThingCustomerId": null,
        "ThingLocationId": null,
        "DYN_ENT_core_cooler__Product": {
        "results": [
            {
                "ObjectType": "core.cooler.package:Product",
                "ObjectId": "FD8EE8B1A322462689C4E4ACC2CEB3DB",

```

```

"Marinda",
"Marinda",
Date(1558616456000)/",
Date(1563969656000)/"
    }
    },
    "DYN_ENT_core_cooler__Address": {
    "results": [
        {
            "ObjectType": "core.cooler.package:Address",
            "ObjectId": "24117A3D559B4153A0183A50A818B124",
            "core.cooler.package.Address.Country": "India",
            "core.cooler.package.Address.City": "Delhi",
            "core.cooler.package.Address.StartTime": "/",
            "core.cooler.package.Address.ExpiryTime": "/"
        }
    ]
}

```

# 11 Actions

An action is a system activity that can be triggered by a rule or another action.

Actions are used to model the system activities that shall be performed when a rule condition is fulfilled or when another action is configured to trigger further actions (sometimes referred to as an action chain). Actions serve as the link between highly technical IoT scenarios focused on sensor data on the one hand and classical business scenarios on the other hand, such as triggering a service request, ordering a spare part, or purchasing an asset as a replacement for an existing one. Also, for more complex scenarios, you can configure an action to trigger an entire workflow containing a predefined sequence of processing steps. Actions can also be triggered by decision support systems as recommended actions that have been prepared for handling certain business situations.

## Prerequisites

For execution of actions that are configured for triggering activities or processes in remote systems, the user executing the action needs the relevant authorizations in the remote system.

### Base URI:

- Formal description: `https://<server address>[:<port number>]/[path]/v1/Actions`
- Example for a base URI in a cloud foundry environment: `https://business-partner.cfapps.eu10.hana.ondemand.com/v1/Actions`

## Methods

HTTP Method	Action	URI	Scopes
<i>POST</i>	<a href="#">Create an Action [page 689]</a>	/Actions	<bp>.c <bp>.r <bp>.tenant.r
<i>GET</i>	<a href="#">Read an Action [page 692]</a>	/Actions ('<ID>')	<bp>.r <bp>.tenant.r
<i>GET</i>	<a href="#">Read all Actions [page 694]</a>	/Actions	<bp>.r <bp>.tenant.r

HTTP Method	Action	URI	Scopes
<i>PUT</i>	<a href="#">Update an Action [page 691]</a>	/Actions ('<ID>')	<bp>.r <bp>.u <bp>.tenant.r
<i>DELETE</i>	<a href="#">Delete Actions [page 697]</a>	/Actions ('<ID>')	<bp>.d <bp>.r <bp>.tenant.r

## 11.1 Create an Action

This service is used to create an action.

### Request

**URI:** /v1/actions

**HTTP Method:** *POST*

**Authorization:** OAuth

**Request Example:** /v1/actions

### Request Headers

Header	Required	Values
Authorization	Yes	Bearer <access-token>
Content-Type	Yes	application/json

### Request Body

#### Payload

```
{
  "actionRunResults": [],
  "actionType": "EMAIL",
  "eventName": null,
  "eventType": "com.sap.apptot.eventtypes:RulesFiredEventType",
  "eventGUID": "4043",
  "name": "test public API",
  "description": "EMAIL",
  "thingTypeGUID": "iotae.automobiles:ABCXSeries",
  "status": 1,
```

```

    "severity": 2,
    "methodType": null,
    "authType": null,
    "destinationName": null,
    "triggeredEventOutbound": false,
    "outboundEventName": null,
    "respDocType": null,
    "invocationType": "AUTO",
    "header": "test",
    "body": "test",
    "requestBody": null,
    "notificationTemplateId": null,
    "notificationTypeKey": null,
    "notificationTypeVersion": null,
    "notificationLanguage": null,
    "notificationTemplateContent": null,
    "url": "",
    "actionParameters": [
      {
        "name": "thing",
        "path": "thing.id",
        "type": "THING",
        "namedPst": null
      },
      {
        "name": "thing",
        "path": "thing.description",
        "type": "THING",
        "namedPst": null
      }
    ],
    "actionRecipients": [
      "<email>"
    ]
  }

```

## Response

Action created. Guid is: "<guid>"

### Response Status and Error Codes

Code	Description
201 Created	Action created
400 Bad Request	Action was empty/Guid not allowed during action creation
401 Unauthorized	Authorization details are invalid/ expired/ not provided
500 Internal Server	Action save failed

## 11.2 Update an Action

This service is used to update certain attributes associated with an action.

### Request

**URI:** /v1/actions

**HTTP Method:** *PUT*

**Authorization:** OAuth

**Request Example:** /v1/actions

### Request Headers

Header	Required	Values
Authorization	Yes	Bearer <access-token>
Content-Type	Yes	application/json

### Request Body

### Payload

```
{
  "guid": "<guid>",
  "actionRunResults": [],
  "actionType": "EMAIL",
  "eventName": null,
  "eventType": "com.sap.apptot.eventtypes:RulesFiredEventType",
  "eventGUID": "4043",
  "name": "test public API",
  "description": "EMAIL",
  "thingTypeGUID": "iotae.automobiles:ABCXSeries",
  "status": 1,
  "severity": 2,
  "methodType": null,
  "authType": null,
  "destinationName": null,
  "triggeredEventOutbound": false,
  "outboundEventName": null,
  "respDocType": null,
  "invocationType": "AUTO",
  "header": "test",
  "body": "test",
  "requestBody": null,
  "notificationTemplateId": null,
  "notificationTypeKey": null,
  "notificationTypeVersion": null,
  "notificationLanguage": null,
  "notificationTemplateContent": null,
  "url": "",
  "actionParameters": [
    {
```

```

    "name": "thing",
    "path": "thing.id",
    "type": "THING",
    "namedPst": null
  },
  {
    "name": "thing",
    "path": "thing.description",
    "type": "THING",
    "namedPst": null
  }
],
"actionRecipients": [
  "<email>"
]
}

```

## Response

Action saved. Guid is: "<guid>"

### Response Status and Error Codes

Code	Description
202 Accepted	Action saved
400 Bad Request	Action was empty
401 Unauthorized	Authorization details are invalid/ expired/ not provided
500 Internal Server	Action Guid is missing

## 11.3 Read an Action

This service is used to retrieve the details of a specified action.

### Request

**URI:** /v1/actions

**HTTP Method:** *GET*

**Authorization:** OAuth

**Request Example:** /v1/actions?guid=<guid>



## Request Parameters

Parameter	Required	Data Type	Description
guid	Yes	String	Internal identifier for action

## Request Headers

Header	Required	Values
Authorization	Yes	Bearer <access-token>

## Response

```
{
  "hasChildren": false,
  "action": {
    "createdBy": null,
    "createdTime": "2019-05-07T17:01:11.897Z",
    "modifiedBy": null,
    "modifiedTime": "2019-05-07T17:01:11.897Z",
    "guid": "69e66dbbadd5434cb4c7e558381659b0",
    "name": "EMAIL",
    "actionType": "EMAIL",
    "description": "EMAIL",
    "thingTypeGUID": "iotae.automobiles:ABCXSeries",
    "status": 0,
    "severity": "2",
    "invocationType": "AUTO",
    "notificationTemplateId": "5A4DA20F4396442EBF641B029C32F10D",
    "notificationTypeKey": "ACTION_10a33e3ae7b940129087552ff8f9f27d",
    "notificationTypeVersion": "1",
    "notificationLanguage": "EN",
    "notificationTemplateContent": null,
    "eventName": null,
    "eventType": "com.sap.apptot.eventtypes:RulesFiredEventType",
    "eventGUID": "4031",
    "url": "",
    "methodType": null,
    "authType": null,
    "requestBody": null,
    "destinationName": null,
    "respDocType": null,
    "triggeredEventOutbound": false,
    "outboundEventName": null,
    "header": "test",
    "body": "test",
    "destinationDetails": null,
    "tenantId": null,
    "cacheInstruction": null,
    "actionParameters": [
      {
        "createdBy": null,
        "createdTime": "2019-05-07T17:01:11.915Z",
        "modifiedBy": null,
        "modifiedTime": "2019-05-07T17:01:11.915Z",
        "guid": "a9f39237bdd941c8a3cdd701a13502a6",
        "actionGUID": "69e66dbbadd5434cb4c7e558381659b0",
        "type": "MD",
        "path": "iotae.automobiles:ABCXSeries:Brand",
        "name": "Brand",
```

```

        "namedPst": null,
        "eventTriggerGUID": null
    },
    ],
    "actionRecipients": [
        {
            "createdBy": null,
            "createdTime": "2019-05-07T17:01:11.916Z",
            "modifiedBy": null,
            "modifiedTime": "2019-05-07T17:01:11.916Z",
            "guid": "096390e0562f437fa09ab8d5e620c579",
            "recipientId": "<email>",
            "actionGUID": "69e66dbbadd5434cb4c7e558381659b0"
        }
    ],
    "actionRunResults": [],
    "targetObject": null,
    "targetAction": null,
    "actionTargetParameters": [],
    "messagingTargetType": null,
    "messagingTargetName": null
},
"missingProps": []
}

```

## Response Status and Error Codes

Code	Description
200 OK	Response
404 Not Found	Action ID not found
401 Unauthorized	Authorization details are invalid/ expired/ not provided
500 Internal Server	Retrieval of action failed

## 11.4 Read all Actions

This service is used to retrieve all the existing actions.

### Request

**URI:** /v1/actions

**HTTP Method:** *GET*

**Authorization:** OAuth

**Request Example:** /v1/actions

### Request Parameters

None

## Request Headers

Header	Required	Values
Authorization	Yes	Bearer <access-token>

## Response

```
[
  {
    "createdBy": null,
    "createdTime": "2019-04-25T23:03:10.550Z",
    "modifiedBy": null,
    "modifiedTime": "2019-04-25T23:56:21.998Z",
    "guid": "<guid>",
    "name": "Service Ticket for low health level battery",
    "actionType": "EMAIL",
    "description": "create Service Ticket for low health level battery",
    "thingTypeGUID": "iotae.automobiles:ABCXSeries",
    "status": 0,
    "severity": null,
    "invocationType": "AUTO",
    "notificationTemplateId": null,
    "notificationTypeKey": null,
    "notificationTypeVersion": null,
    "notificationLanguage": null,
    "notificationTemplateContent": null,
    "eventName": null,
    "eventType": "com.sap.apptot.eventtypes:RulesFiredEventType",
    "eventGUID": "em-1",
    "url": null,
    "methodType": "POST",
    "authType": null,
    "requestBody": null,
    "respDocType": "JSON",
    "triggeredEventOutbound": false,
    "outboundEventName": null,
    "header": null,
    "body": null,
    "destinationDetails": null,
    "tenantId": null,
    "cacheInstruction": null,
    "actionParameters": [],
    "actionRecipients": [],
    "actionRunResults": [],
    "targetObject": null,
    "targetAction": null,
    "actionTargetParameters": [],
    "messagingTargetType": null,
    "messagingTargetName": null
  },
  {
    "createdBy": null,
    "createdTime": "2019-04-16T23:48:25.495Z",
    "modifiedBy": null,
    "modifiedTime": "2019-04-16T23:53:46.777Z",
    "guid": "<guid>",
    "name": "email",
    "actionType": "EMAIL",
    "description": "create Service Ticket for low health level battery",
    "thingTypeGUID": "iotae.automobiles:ABCXSeries",
    "status": 0,
  }
]
```

```

    "severity": null,
    "invocationType": "AUTO",
    "notificationTemplateId": "TEST",
    "notificationTypeKey": "ACTION_1",
    "notificationTypeVersion": "1",
    "notificationLanguage": "EN",
    "notificationTemplateContent": null,
    "eventName": null,
    "eventType": "com.sap.appiot.eventtypes:RulesFiredEventType",
    "eventGUID": "email1",
    "url": null,
    "methodType": null,
    "authType": null,
    "requestBody": null,
    "destinationName": null,
    "respDocType": null,
    "triggeredEventOutbound": false,
    "outboundEventName": null,
    "header": "Test",
    "body": "Low fillLevel",
    "destinationDetails": null,
    "tenantId": null,
    "cacheInstruction": null,
    "actionParameters": [],
    "actionRecipients": [],
    "actionRunResults": [],
    "targetObject": null,
    "targetAction": null,
    "actionTargetParameters": [],
    "messagingTargetType": null,
    "messagingTargetName": null
  }
]

```

## Response Status and Error Codes

Code	Description
200 OK	Response
404 Not Found	Action ID not found
401 Unauthorized	Authorization details are invalid/ expired/ not provided
500 Internal Server	Retrieval of actions failed

## 11.5 Delete an Action

This service is used to delete an action with a specific action ID.

### Request

**URI:** `/v1/actions`

**HTTP Method:** `DELETE`

**Authorization:** OAuth

**Request Example:** /v1/actions?guid=<guid>

### Request Parameters

Parameter	Required	Data Type	Description
guid	Yes	String	Internal identifier for action

### Request Headers

Header	Required	Values
Authorization	Yes	Bearer <access-token>
Content-Type	Yes	application/json

## Response

### Response Status and Error Codes

Code	Description
204 No Content	Success
404 Not Found	Action not found
401 Unauthorized	Authorization details are invalid/ expired/ not provided
500 Internal Server	Deletion of action failed

## 11.6 Delete Actions

This service is used to delete list of actions.

### Request

**URI:** /v1/actions

**HTTP Method:** *DELETE*

**Authorization:** OAuth

**Request Example:** /v1/actions

### Request Parameters

None

## Request Headers

Header	Required	Values
Authorization	Yes	Bearer <access-token>
Content-Type	Yes	application/json

## Request body

```
[ "<guid>", "<guid>" ]
```

## Response

### Response Status and Error Codes

Code	Description
204 No Content	Success
401 Unauthorized	Authorization details are invalid/ expired/ not provided
500 Internal Server	Deletion of action failed

## 11.7 Create an Action Event

This service is used to create an action event.

## Request

**URI:** /v1/action/event

**HTTP Method:** *POST*

**Authorization:** OAuth

**Request Example:** /v1/action/event

### Request Headers

Header	Required	Values
Authorization	Yes	Bearer <access-token>

Header	Required	Values
Content-Type	Yes	application/json

## Request Body

### Payload

```
{
  "cloudEventsVersion": "0.1",
  "eventID": "<eventId>",
  "eventType": "com.sap.apptot.eventtypes:RulesFiredEventType",
  "eventTime": "2019-05-13T08:50:00.000Z",
  "contentType": "application/json",
  "data": {
    "ruleID": "<ruleId>",
    "thingId": "<thingId>",
    "ingestionTime": "2019-05-10T08:50:00.000Z",
    "source": []
  },
  "extensions": {
    "type": "Alert",
    "eventInfo": "NewEvent",
    "eventStatus": "Open",
    "eventSeverity": 1,
    "eventCode": "",
    "eventSource": "19999",
    "externalID": "<externalID>",
    "correlationID": "test"
  }
}
```

## Response

Event has been handed over to the Kafka producer.

### Response Status and Error Codes

Code	Description
200 OK	Event has been handed over to the Kafka producer
400 Bad Request	Required request body is missing
401 Unauthorized	Authorization details are invalid/ expired/ not provided

## 11.8 Update an Action Event

This service is used to update an action event.

### Request

URI: `/v1/actionEvent`

HTTP Method: *PUT*

Authorization: OAuth

Request Example: `/v1/actionEvent`

### Request Headers

Header	Required	Values
Authorization	Yes	Bearer <access-token>
Content-Type	Yes	application/json

### Request Body

#### Payload

```
{"status": "Approved", "eventId": "<eventId>"}
```

### Response

#### Response Status and Error Codes

Code	Description
202 Accepted	Received and published/ Received but not published due to status
401 Unauthorized	Authorization details are invalid/ expired/ not provided



## 11.9 Get Pending/all Action Events

This service is used to get the pending or all the action events.

### Request

**URI:** /v1/actionEvent

**HTTP Method:** *GET*

**Authorization:** OAuth

**Request Example:** /v1/actionEvent?\$filter=pending /v1/actionEvent?\$filter=all

### Request Parameters

Parameter	Required	Data Type	Description
\$filter=all	No	String	List of all action events
\$filter=pending	No	String	List of all pending action events

### Request Headers

Header	Required	Values
Authorization	Yes	Bearer <access-token>
Content-Type	Yes	application/json

### Response

```
[
  {
    "createdBy": null,
    "createdTime": "2019-02-21T00:24:24.822Z",
    "modifiedBy": null,
    "modifiedTime": "2019-02-21T00:28:19.495Z",
    "eventId": "<eventId>",
    "eventTime": "2019-02-21T00:23:00.808+0000",
    "eventType": "com.sap.apptot.eventtypes:RulesFiredEventType",
    "sourceGUID": "3000",
    "eventContent": "{\n  \"cloudEventsVersion\": \"0.1\",\n  \"eventID\": \"3000\",\n  \"eventType\": \"com.sap.apptot.eventtypes:RulesFiredEventType\",\n  \"eventTime\": \"2018-12-12T17:30:00Z\",\n  \"contentType\": \"application/json\",\n  \"data\": {\n    \"ruleID\": \"3000\",\n    \"thingID\": \"<thingID>\",\n    \"ingestionTime\": \"2018-12-12T18:00:00Z\",\n    \"source\": [\n      {\n        \"propertySet\": \"core.automobiles:Engine\",\n        \"properties\": [\n          {\n            \"objectName\": \"Temperature\",\n            \"objectValue\": \"200\"\n          },\n          {\n            \"objectName\": \"RPM\",\n            \"objectValue\": \"3000\"\n          }\n        ]\n      }\n    ],\n    \"extensions\": {\n      \"type\": \"Alert\",\n      \"eventInfo\": \"Alert on temperature\",\n      \"eventStatus\": \"Open\",\n      \"eventSeverity\": 1,\n      \"eventCode\": null,\n      \"eventSource\": null,\n      \"externalID\": \"Engine_0012\",\n      \"correlationID\": \"E6\"\n    }\n  }\",
```

```

"status": "APPROVED",
"action": {
  "createdBy": null,
  "createdTime": "2019-05-09T04:46:50.449Z",
  "modifiedBy": null,
  "modifiedTime": "2019-05-09T04:54:52.608Z",
  "actionParameters": [
    {
      "createdBy": null,
      "createdTime": "2019-05-09T04:54:52.035Z",
      "modifiedBy": null,
      "modifiedTime": "2019-05-09T04:54:52.035Z",
      "actionGUID": "<actionGUID>",
      "type": "THING",
      "path": "thing.type",
      "name": "thing",
      "namedPst": null,
      "eventTriggerGUID": null,
      "guid": "<guid>"
    }
  ],
  "actionRecipients": [
    {
      "createdBy": null,
      "createdTime": "2019-05-09T04:54:52.036Z",
      "modifiedBy": null,
      "modifiedTime": "2019-05-09T04:54:52.036Z",
      "recipientId": "<email>",
      "actionGUID": "<actionGUID>",
      "guid": "<guid>"
    }
  ],
  "actionRunResults": null,
  "actionTargetParameters": [],
  "actionType": "EMAIL",
  "eventName": null,
  "eventType": "com.sap.apptot.eventtypes:RulesFiredEventType",
  "eventGUID": "<eventGUID>",
  "name": "test",
  "description": "EMAIL",
  "thingTypeGUID": "iotae.automobiles:ABCXSeries",
  "status": 0,
  "severity": "2",
  "methodType": null,
  "authType": null,
  "destinationName": null,
  "triggeredEventOutbound": false,
  "outboundEventName": null,
  "respDocType": null,
  "invocationType": "AUTO",
  "header": "test",
  "body": "test",
  "requestBody": null,
  "notificationTemplateId": "test",
  "notificationTypeKey": "ACTION_1",
  "notificationTypeVersion": "1",
  "notificationLanguage": "EN",
  "targetObject": null,
  "targetAction": null,
  "messagingTargetType": null,
  "messagingTargetName": null,
  "destinationDetails": null,
  "tenantId": null,
  "notificationTemplateContent": null,
  "cacheInstruction": null,
  "url": "",
  "guid": "<guid>"
},

```

```

    "guid": "<guid>"
  },
  {
    "createdBy": null,
    "createdTime": "2019-02-21T00:24:24.822Z",
    "modifiedBy": null,
    "modifiedTime": "2019-02-21T00:28:19.495Z",
    "eventId": "<eventId>",
    "eventTime": "2019-02-21T00:23:00.808+0000",
    "eventType": "com.sap.apptot.eventtypes:RulesFiredEventType",
    "sourceGUID": "3000",
    "eventContent": "{\n  \"cloudEventsVersion\": \"0.1\", \"eventID\": \"3000\",
    \"eventType\": \"com.sap.apptot.eventtypes:RulesFiredEventType\", \"eventTime\":
    \"2018-12-12T17:30:00Z\", \"contentType\": \"application/json\", \"data\": {\n    \"ruleID
    \": \"3000\", \"thingID\": \"<thingID>\", \"ingestionTime\": \"2018-12-12T18:00:00Z\",
    \"source\": [\n      {\n        \"propertySet\": \"core.automobiles:Engine\", \"properties\":
        [\n          {\n            \"objectName\": \"Temperature\", \"objectValue\": \"200\"
          }, {\n            \"objectName\": \"RPM\", \"objectValue\": \"3000\"
          }
        ]
      }, {\n        \"type\": \"Alert\", \"eventInfo\": \"Alert on temperature\",
        \"eventStatus\": \"Open\", \"eventSeverity\": 1, \"eventCode\": null,
        \"eventSource\": null, \"externalID\": \"Engine_0012\", \"correlationID\": \"E6\"
      }
    ]
  }",
    "status": "APPROVED",
    "action": {
      "createdBy": null,
      "createdTime": "2019-05-09T04:46:50.449Z",
      "modifiedBy": null,
      "modifiedTime": "2019-05-09T04:54:52.608Z",
      "actionParameters": [
        {
          "createdBy": null,
          "createdTime": "2019-05-09T04:54:52.035Z",
          "modifiedBy": null,
          "modifiedTime": "2019-05-09T04:54:52.035Z",
          "actionGUID": "<actionGUID>",
          "type": "THING",
          "path": "thing.type",
          "name": "thing",
          "namedPst": null,
          "eventTriggerGUID": null,
          "guid": "<guid>"
        }
      ]
    },
    "actionRecipients": [
      {
        "createdBy": null,
        "createdTime": "2019-05-09T04:54:52.036Z",
        "modifiedBy": null,
        "modifiedTime": "2019-05-09T04:54:52.036Z",
        "recipientId": "<email>",
        "actionGUID": "<actionGUID>",
        "guid": "<guid>"
      }
    ],
    "actionRunResults": null,
    "actionTargetParameters": [],
    "actionType": "EMAIL",
    "eventName": null,
    "eventType": "com.sap.apptot.eventtypes:RulesFiredEventType",
    "eventGUID": "<eventGUID>",
    "name": "test",
    "description": "EMAIL",
    "thingTypeGUID": "iotae.automobiles:ABCXSeries",
    "status": 0,
    "severity": "2",
    "methodType": null,
    "authType": null,
    "destinationName": null,
    "triggeredEventOutbound": false,

```

```

    "outboundEventName": null,
    "respDocType": null,
    "invocationType": "AUTO",
    "header": "test",
    "body": "test",
    "requestBody": null,
    "notificationTemplateId": "test",
    "notificationTypeKey": "ACTION_1",
    "notificationTypeVersion": "1",
    "notificationLanguage": "EN",
    "targetObject": null,
    "targetAction": null,
    "messagingTargetType": null,
    "messagingTargetName": null,
    "destinationDetails": null,
    "tenantId": null,
    "notificationTemplateContent": null,
    "cacheInstruction": null,
    "url": "",
    "guid": "<guid>"
  },
  "guid": "<guid>"
}]

```

## Response Status and Error Codes

Code	Description
202 Accepted	Received and published/ Received but not published due to status
401 Unauthorized	Authorization details are invalid/ expired/ not provided

# 12 Thing

The thing services allow you to model objects that you created using the configuration service. You can create a thing instance which constitutes to an individual tangible object.

IoT applications usually deal with very high volumes of data. To be able to handle data in a uniform and cost-effective way, the thing service consists of two main parts:

- Storage and retrieval of the thing configuration  
This corresponds to the thing types and property set type details.
- Storing and reading data specific to a thing based on the definitions in the thing configuration  
This corresponds to services such as creating data for a thing, and reading data such as properties, time series and so on.

## i Note

- The thing configuration services return thing configuration information from the packages –
- within the same tenant to which a user belongs, only if the scope defines the package as private

## Access Authorization for Things

Objects of type `Thing` are subject to the instance-based authorization concept that is implemented in SAP Leonardo IoT. For more information, see [Object Instance-based Authorization \[page 205\]](#). To grant access to a thing, a capability has to be derived from the predefined `CapabilityType` object for things. For more information on the `CapabilityType` payload needed for this purpose, see [Payload of CapabilityType "Thing" \[page 746\]](#).

## Related Information

[Onboarding Things \[page 715\]](#)

[Thing Data \[page 705\]](#)

## 12.1 Thing Data

Overview of various data retrieval scenarios for thing data

After creating a thing instance, you can store and retrieve data for properties specific to a thing based on the definitions in the thing configuration. The varied type of data stored for the properties depends on the data categories such as [MasterData](#), [TimeSeriesData](#), [Parameters](#), [ReferencePropertyData](#), and [DerivedData](#) defined in the configuration. In this section, you will find the list of services offered by SAP Leonardo IoT to retrieve data in different scenarios – master data history, aggregates, measurements, and composite data are to name a few. You can choose one of the services to retrieve data according to your business needs.

What you want to know?	Which service to choose?	What the service does?
The prior owners and the prior license plate details of a specific car  (Historical data of a thing)	<a href="#">Read Data for a Thing [page 849]</a>	Reads master data for a combination of thing ID, thing type, and property set within the specified time range.
The latest license plate details of multiple cars  (Snapshot data across multiple things)	<a href="#">Read Property Set Data [page 854]</a>	Reads master data for all property sets of the specified property set type. You can filter the result based on a specific thing ID or timestamp.
The latest license plate details along with other basic details, customer details, location details, and event details of all cars  (Snapshot data across multiple things)	<a href="#">Read Thing Details with Property Set Type Data [page 1044]</a>  REST Service	Read the latest master data for the specified combination of property set type name and property set ID. The result also includes the basic details, customer details, and event details.
The latest license plate details along with other basic details, customer details, location details, and event details of all cars  (Snapshot data across multiple things)	<a href="#">Read Things for Property Set of a Property Set Type [page 1058]</a>  OData Service	Read the latest master data for the specified property set type name and filter on the property set ID. The result also includes the basic details, customer details, and event details.
The sensor values for the front wheel of a specific car during 1st week of July 2018  (Historical data of a thing)	<a href="#">Read Time Series Data or Derived Data for a Thing [page 883]</a>	Read time series data for a combination of thing ID, thing type, and property set within the specified time range. The time range must be within the retention period.
The latest value of all properties of a car in January 2017  (Snapshot data of a thing)	<a href="#">Read Snapshot Data Within a Time Range for a Thing [page 906]</a>	Read snapshot of latest master data or time series data for the specified thing ID.
The sensor values for the front wheel of a specific car during last ten days of December 2015  (Historical data of a thing)	<a href="#">Read Time Series Data or Derived Data for a Thing [page 913]</a>	Read time series data for a combination of thing ID, thing type, and property set within the specified time range. The time range must be outside the retention period.
The average speed per model and year across all cars  (Aggregate data across multiple things)	<a href="#">Read Aggregates of Measurements [page 922]</a>	Reads aggregate measurements for average, median + 10 more aggregates for the specified property set type, grouped by time or other dimension properties within a time range.
The sensor values for wheels of multiple cars in the first half of 2018  (Historical data across multiple things)	<a href="#">Read Multiple Measurements [page 933]</a>	Read time series data for all things that uses a specific property set type. You can filter the data by thing ID and time range.
The location of the car, the pressure value of the front wheel of the car, and an aggregate of events raised on the sensor values of the car  (Snapshot data of a thing)	<a href="#">Read Property Set Data and Events for a Thing [page 1162]</a>	Read master data, time series data, and events of a specific thing ID. You can filter the result based on the thing type, property set type, and event type by specifying one or more of them as request header parameters.

What you want to know?	Which service to choose?	What the service does?
Extreme sensor values of the car in December 2017 with the split up of the time range into 31 segments  (Aggregate data for a thing)	<a href="#">Read Time Series Data for Multiple Property Types of a Thing Applying M4 Algorithm [page 897]</a>	Reads the sensor values for a combination of thing ID, thing type, and multiple properties for a specific number of divisions within the specified time range. The service returns unique minimum value, maximum value, first occurring value, and last occurring value.
The unit of measure values and the threshold limits for the wheel of a car	<a href="#">Read Data for Reference Properties [page 862]</a>	Reads the reference data such as unit of measure and threshold values available for the last one hour for the specified property set of a thing type .

## Related Information

[Time Series Data Ingestion \[page 707\]](#)

[Thing: Master Data and Parameters Data \[page 840\]](#)

[Thing: Reference Property Data \[page 858\]](#)

## 12.2 Time Series Data Ingestion

Time series data ingestion in SAP Leonardo IoT covers the following:

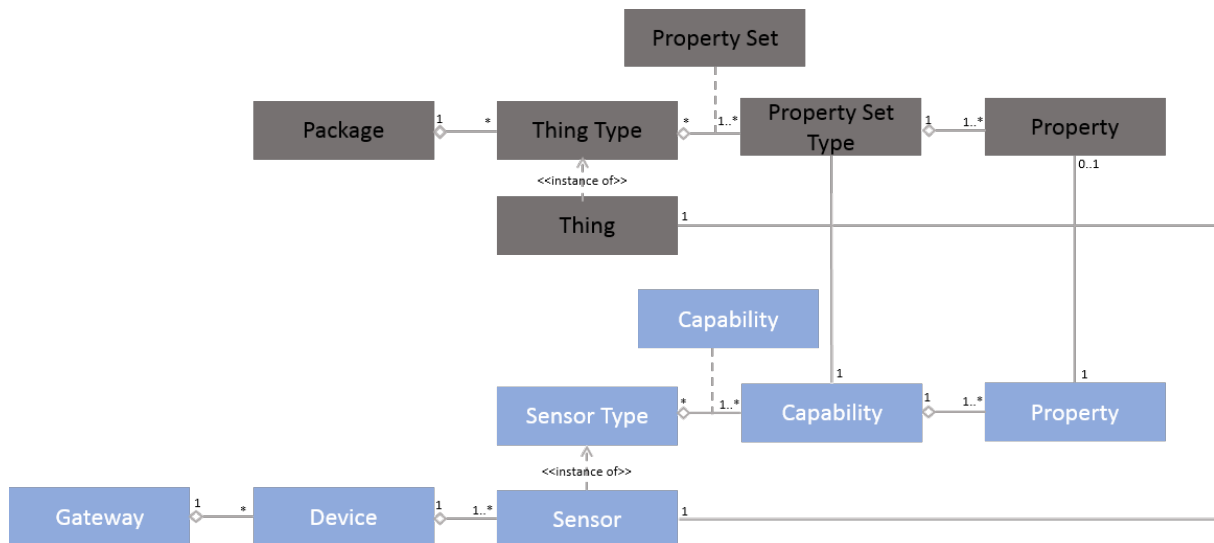
- Integration with device connectivity components to define device models
- Map the device models with SAP Leonardo IoT thing model
- Receive sensor data from the devices
- Convert the raw data into time series data
- Store the time series data in SAP Leonardo IoT Time Series Store

Speaking of things in the Internet of Things only makes sense if you extend the focus to the sensors that are attached to the things you want to control or monitor in an IoT scenario. It is the sensors that are the very source of the data that can be collected in the form of timeseries data. Here is how the various objects involved come together during thing onboarding.

### Data Model Mapping

During the first iteration, the device model of IoT service will be mapped to the Thing Model as follows:

- A *Sensor* is mapped to a *Thing* via the *sensorId*.
- A *Capability* is mapped to a *PropertySetType* via the *capabilityId*.
- A *Property* to a *Property* via the *propertyId*.



## Thing Onboarding

When setting up an IoT scenario, one major step is the so-called onboarding. This term refers to the process of establishing a relationship between a physical entity and its virtual counterpart known as the digital twin. The physical entities that can be onboarded comprise the following:

- Persons
- Organizations
- Things

While onboarding of organizations and persons is covered by the Tenant Administration apps, the onboarding of things is a task that you perform with the Thing Modeler app. The onboarding process with the Tenant Administration apps basically means setting up a virtual mirror object for a real-world entity. In contrast to that, onboarding a thing does not only set up such a mirror object. Rather, a technical connection between a physical device and its digital twin is established. This connection is then used for transferring sensor data from a device into the IoT storage system for monitoring and analytics.

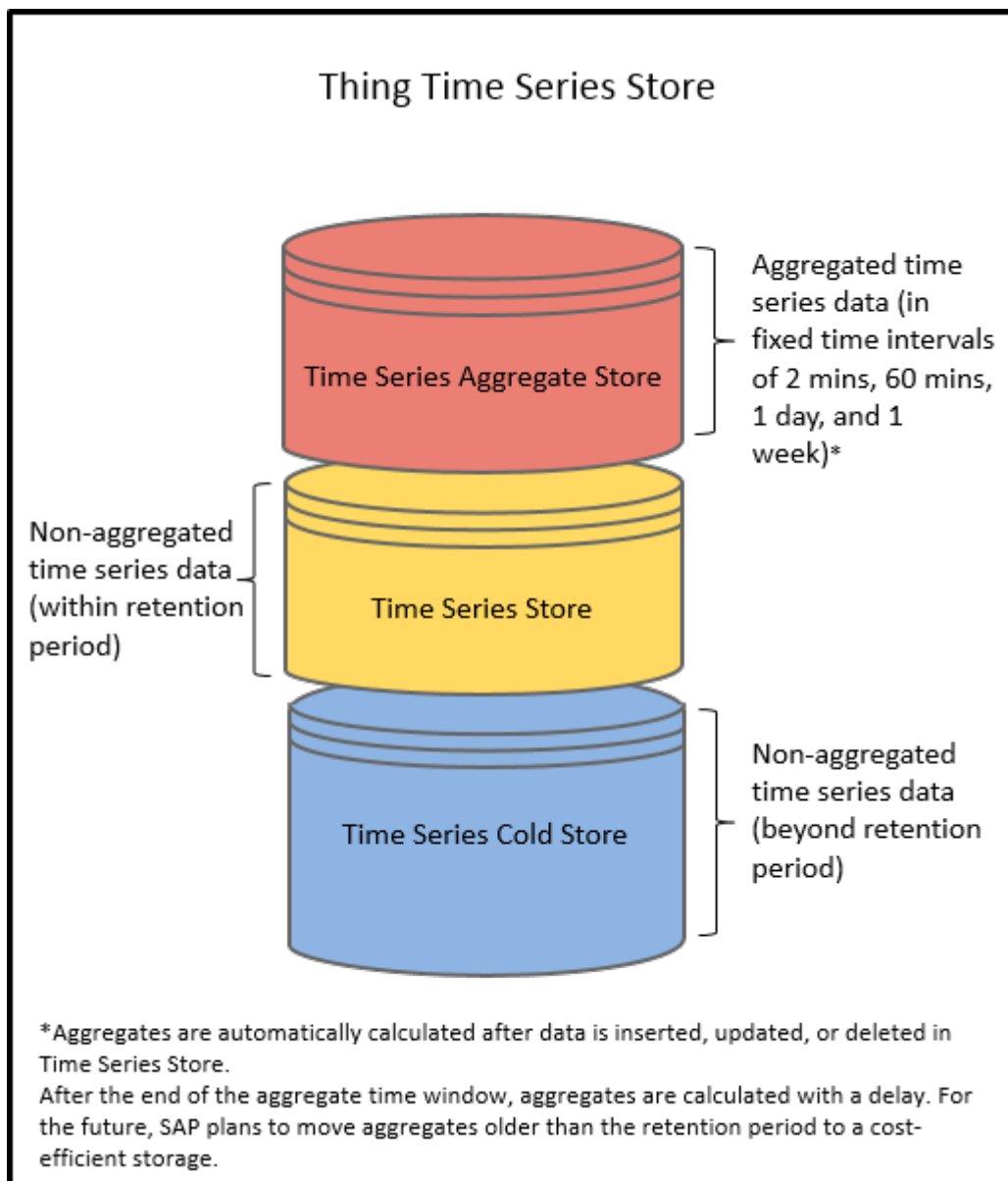
## Thing Time Series Store

The Thing Time Series Store allows you to initialize the time series store for a tenant, manage retention period for a tenant, and manage time series data for Things belonging to a tenant. The time series store uses a tiered storage model containing the following tiers:

- Time Series Store
- Time Series Aggregate Store
- Time Series Cold Store



The following diagram illustrates the Thing Time Series Store:



You can create time series data for a thing using the REST-based API [Create Time Series Data or Derived Data for a Thing \[page 877\]](#) or you can use SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (Internet of Things Service) for importing time series data into SAP Leonardo IoT by using the data ingestion services provided by the SAP Leonardo IoT API.

### Related Information

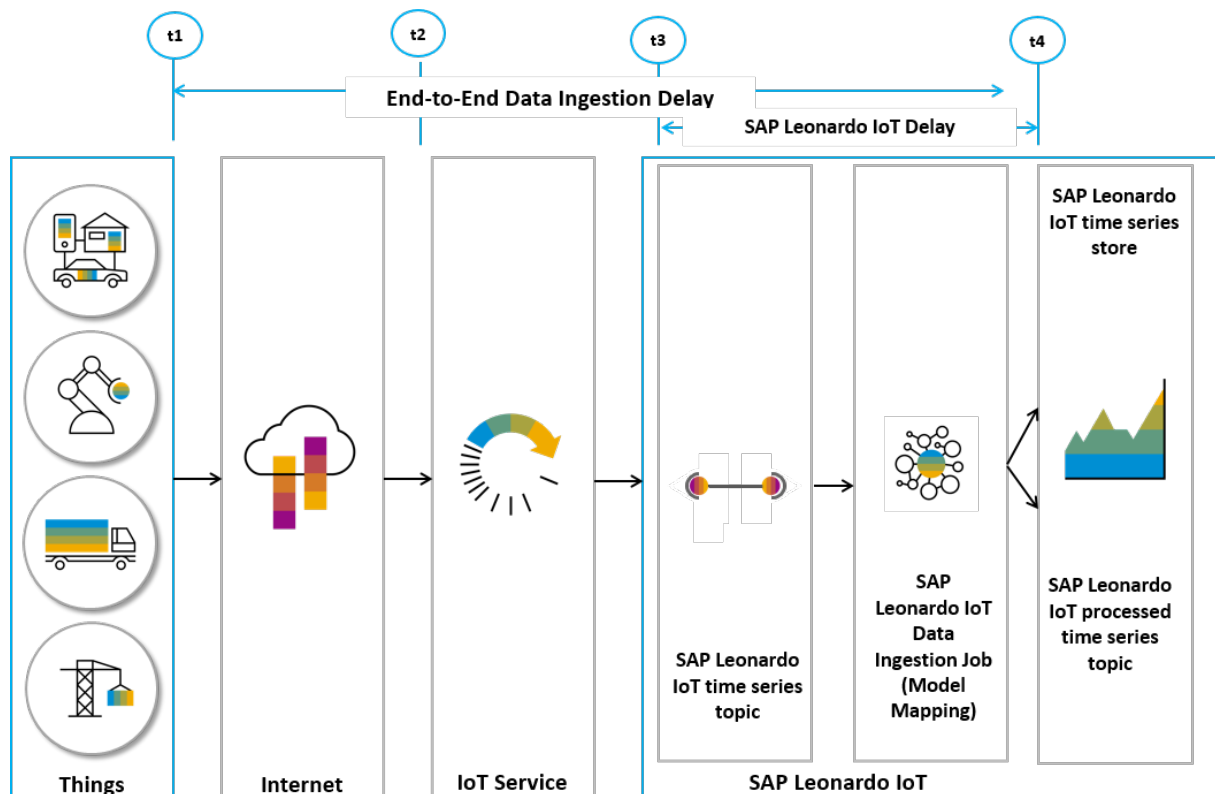
- [Sensors and Things \[page 748\]](#)
- [Thing Onboarding](#)
- [Thing: Time Series Data \[page 870\]](#)
- [Monitoring Time Series Data Ingestion Delay \[page 710\]](#)

## 12.2.1 Monitoring Time Series Data Ingestion Delay

### Overview

The sensor data from the physical devices traverses different layers such as device management system (for example, SAP Cloud Platform Internet of Things for the Cloud Foundry Environment), virtual things modeled with SAP Leonardo IoT, and is finally persisted in the Time Series Store. Monitoring the traversal of data helps the customer analyze the end-to-end ingestion delay – in other words, the time from when the data was emitted by the sensor until it was persisted in the Time Series Store.

The following illustration helps understand the different types of delay in data ingestion:



1. **Time t1:** A device sensor emits data.
2. **Time t2:** The time series data from sensor is transmitted over the internet and routed to SAP Cloud Platform Internet of Things for the Cloud Foundry Environment via REST call or MQTT protocol. The sensor device is identified, authenticated, and the data is validated.
3. **Time t3:** SAP Leonardo IoT receives time series data from IoT Services.
4. **Time t4:** After processing the data, it is stored in the time series store.

### Data Ingestion Delays

SAP Leonardo IoT provisions the following data ingestion delays:

**End-to-End Data Ingestion Delay (End\_to\_end\_Delay)** – Time between the business timestamp for a measurement sent by the sensor and the time when the measurement is stored in the Time Series Store. For example, in the above illustration the difference between time **t4** and time **t1** indicates the end-to-end data ingestion delay.

**SAP Leonardo IoT Delay (IoTAE\_Ingest\_Delay)** – Time between the timestamp when a measurement is ingested into SAP Leonardo IoT and the time when the measurement is stored in the Time Series Store. For example, in the above illustration the difference between time **t4** and **t3** indicates the SAP Leonardo IoT delay.

**\_time** – The time stamp when SAP Leonardo IoT initiated the data ingestion delay calculation.

**LastBusinessTimestampProcessed** – Latest **\_time** value in the sample set of ingested measurements. The value in this field helps to understand which ingested measurement was used for calculating the ingestion delay. For example, from the following random sample set of ingested measurements, the random **\_time** value 1544086800000 (that is, 2018-12-06T09:00:00) is used for calculating the ingestion delay.

```
{
  "capabilityAlternateId": 19,
  "sensorAlternateId": "124223",
  "measures": [{
    "temperature": "27.63",
    "_time": 1544086800000
  }]
}
```

```
{
  "capabilityAlternateId": 19,
  "sensorAlternateId": "124223",
  "measures": [{
    "temperature": "28.63",
    "_time": 1544086802000
  }]
}
```

```
{
  "capabilityAlternateId": 19,
  "sensorAlternateId": "124223",
  "measures": [{
    "temperature": "25.23",
    "_time": 1544086804000
  }]
}
```

```
{
  "capabilityAlternateId": 19,
  "sensorAlternateId": "124223",
  "measures": [{
    "temperature": "29.63",
    "_time": 1544086806000
  }]
}
```

```
{
  "capabilityAlternateId": 19,
  "sensorAlternateId": "124223",
  "measures": [{
    "temperature": "20.63",
    "_time": 1544086808000
  }]
}
```

The delays are calculated during the ingestion process for a specific time interval, which is configurable and is one minute by default. However, in the following scenarios, the SAP Leonardo IoT delay will not be updated for the specified time interval:

- Data is not ingested through SAP Cloud Platform Internet of Things for the Cloud Foundry Environment

- Ingested data is not processed
- Ingestion processing time is more than a minute

## Enabling Data Ingestion Delay Monitoring

You must create a support ticket under the component `IOT-AE-BDI` for monitoring the data ingestion delay. In the support ticket, you must specify the time interval for which the delay will be calculated. The supported time intervals are 30 seconds, one minute, two minutes, and five minutes. By default, the delay is calculated for one minute. After receiving an approval for the ticket, you must create a thing (per tenant) with a predefined alternate thing ID (`_SAPIOT_Monitoring_Thing`). The following is a sample payload to create a thing.

```
{
  "_externalId": "ingestion.monitor",
  "_alternateId": "_SAPIOT_Monitoring_Thing",
  "_name": "Thing",
  "_description": {
    "en": "English"
  },
  "_thingType": [
    "com.sap.iotaestandardcontent:SAPIOT_Monitoring"
  ],
  "_objectGroup": "883BB1AFC4A4425791543V2F2CFDD295"
}
```

For information on how to create a thing, see [Create a Thing \[page 716\]](#).

Once the monitoring is enabled, you can create a monitoring dashboard, for example, using IoT Application project template in SAP Web IDE or monitor the delay details using specific APIs provided by SAP Leonardo IoT.

The following are the API services that will help you retrieve the delay details for the thing you have configured. You must specify the alternate thing ID in the request URL.

### Read Time Series Data API

Request URL: `/Things(' _SAPIOT_Monitoring_Thing')/com.sap.iotaestandardcontent:SAPIOT_Monitoring/SAPIOT_DataIngest`

Response payload:

```
{
  "value": [
    {
      "time": "2018-12-04T09:33:11.945Z",
      "End_to_end_Delay": 3864,
      "IoTAE_Ingest_Delay": 3859,
      "LastBusinessTimestampProcessed": "2018-12-04T09:31:39.415Z"
    },
    {
      "time": "2018-12-04T09:32:11.945Z",
      "End_to_end_Delay": 3446,
      "IoTAE_Ingest_Delay": 3440,
      "LastBusinessTimestampProcessed": "2018-12-04T09:30:59.826Z"
    },
    {
      "time": "2018-12-04T09:31:11.971Z",
      "End_to_end_Delay": 3689,
      "IoTAE_Ingest_Delay": 3609,
      "LastBusinessTimestampProcessed": "2018-12-04T09:29:39.670Z"
    }
  ]
}
```

## M4 Algorithm API

Request URL: /Things('\_SAPIOT\_Monitoring\_Thing')/  
com.sap.iotaestandardcontent:SAPIOT\_Monitoring/SAPIOT\_DataIngest?&  
\$apply=sap.lineChart(divisions=400,property=%27IoTAE\_Ingest\_Delay%27%27)

Response payload:

```
{
  "value": [
    {
      "time": "2018-12-04T09:05:12.011Z",
      "IoTAE_Ingest_Delay": 3429
    },
    {
      "time": "2018-12-04T09:06:12.039Z",
      "IoTAE_Ingest_Delay": 3905
    },
    {
      "time": "2018-12-04T09:07:12.023Z",
      "IoTAE_Ingest_Delay": 3637
    }
  ]
}
```

## Snapshot API

Request URL: /Snapshot(thingId='\_SAPIOT\_Monitoring\_Thing',fromTime='',dataCategory='')

Response payload:

```
{
  "_id": "E782B5429DC1436E96C40A10ED1049BF",
  "_thingType": [
    "com.sap.iotaestandardcontent:SAPIOT_Monitoring"
  ],
  "value": [
    {
      "com.sap.iotaestandardcontent:SAPIOT_Monitoring": [
        {
          "description": "",
          "_propertySetType": {
            "name": "com.sap.iotaestandardcontent:SAPIOTSERVICE",
            "description": ""
          },
          "/SAPIOTSERVICE": [
            {
              "Ingest_Delay": {
                "_unitOfMeasure": "",
                "description": "",
                "value": null,
                "qualityCode": null,
                "time": null
              }
            },
            {
              "IoT_Services_RTT": {
                "_unitOfMeasure": "",
                "description": "",
                "value": null,
                "qualityCode": null,
                "time": null
              }
            }
          ]
        }
      ]
    }
  ],
}
```

```

{
  "/SAPIOT_DataIngest": [
    {
      "End_to_end_Delay": {
        "_unitOfMeasure": "",
        "description": "",
        "value": 3393,
        "qualityCode": null,
        "_time": "2018-12-04T10:09:11.946Z"
      }
    },
    {
      "IoTAE_Ingest_Delay": {
        "_unitOfMeasure": "",
        "description": "",
        "value": 3378,
        "qualityCode": null,
        "_time": "2018-12-04T10:09:11.946Z"
      }
    },
    {
      "LastBusinessTimestampProcessed": {
        "_unitOfMeasure": "",
        "description": "",
        "value": "2018-12-04T10:05:49.889Z",
        "qualityCode": null,
        "_time": "2018-12-04T10:09:11.946Z"
      }
    }
  ],
  "description": "",
  "_propertySetType": {
    "name": "com.sap.iotaestandardcontent:SAPIOT_DataIngest",
    "description": ""
  }
},
{
  "/SAPIOT_Forwarder": [
    {
      "BusinessTimestampProcessed": {
        "_unitOfMeasure": "",
        "description": "",
        "value": null,
        "qualityCode": null,
        "_time": null
      }
    },
    {
      "End_to_end_Delay": {
        "_unitOfMeasure": "",
        "description": "",
        "value": null,
        "qualityCode": null,
        "_time": null
      }
    },
    {
      "isAlive": {
        "_unitOfMeasure": "",
        "description": "",
        "value": null,
        "qualityCode": null,
        "_time": null
      }
    }
  ],
  "description": "",
  "_propertySetType": {

```

```

        "name": "com.sap.iotaestandardcontent:SAPIOT_Forwarder",
        "description": ""
    }
}
]
}
]
}

```

## 12.3 Onboarding Things

A thing is an instance of a thing type. Thing type and thing have a one-to-many relationship, that is, one thing type may be referenced by many things, while each thing is defined by exactly one thing type. For example, you can create a thing of a specific car model ABCXSeries of thing type Automobiles and specify values for the attributes defined in the Automobiles thing type.

You can use the thing services to create, delete, and retrieve things.

### Base URI

- **Formal description:** `http://<server address>[:<port number>][[/path]]/Things`
- Example for a base URI in a cloud foundry environment: `https://apiot-mds.cfapps.eu10.hana.ondemand.com/Things`

### Methods

Model APIs

HTTP Method	Action	URI	Scopes
<i>POST</i>	<a href="#">Create a Thing [page 716]</a>	<code>/Things</code>	<code>&lt;thing&gt;.c</code>
<i>GET</i>	<a href="#">Read a Thing [page 731]</a>	<code>/Things('&lt;thing ID&gt;')</code>	<code>&lt;thing&gt;.r</code>
<i>GET</i>	<a href="#">Read all Things [page 743]</a>	<code>/Things</code>	<code>&lt;thing&gt;.r</code>
<i>POST</i>	<a href="#">Update a Thing [page 739]</a>	<code>/Things('&lt;thing ID&gt;')</code>	<code>&lt;thing&gt;.c</code>
<i>DELETE</i>	<a href="#">Delete a Thing [page 742]</a>	<code>/Things('&lt;thing ID&gt;')</code>	<code>&lt;thing&gt;.d</code> <code>&lt;file&gt;.da</code>

#### i Note

`<thing>` refers to the thing application name.

`<auth>` refers to the authorization application name.

## Related Information

[Create Multiple Things \[page 727\]](#)

### 12.3.1 Create a Thing

With this method, you can create a thing instance using a thing type or a thing template.

When you create a thing using this method, the system generates a unique ID for it. In addition, you can assign an alternate thing ID for a thing instance using this method. You can use the system generated thing ID or the alternate thing ID for all thing-relevant API calls. You cannot change the alternate thing ID after it is assigned to a thing. Thing ID and alternate thing ID have a one-to-one relationship within a specific tenant. To create multiple thing instances, see [Create Multiple Things \[page 727\]](#).

#### Thing Type or Thing Template

You can create a thing instance using a thing type or a thing template. However, the thing template is based on a thing type. The benefit of creating a thing instance using a thing template is that you can pre-populate data for a thing. When you create a thing instance using a thing template, internally the method `POST / Things('<thing ID>')/<thing type name>/<property set ID>` is called. The method copies the property values of the template along with the timestamp to the newly created thing instance. You can later change the pre-populated data for a thing.

#### i Note

We recommend not to specify both thing type and thing template in the request payload.

- If you specify the thing type and a thing template of the same thing type in the request payload, the service creates the thing and assigns values for its properties provided in the thing template.
- However, if the thing template does not belong to the same thing type specified in the request payload, the service displays an error message and the thing is not created.

#### Thing Template or Thing Data

When you create a thing, you can assign values for properties that belong to the property set type of data categories [MasterData](#) and [Parameters](#) without using the thing template. You can achieve this by including a separate data block in the request payload.

#### i Note

We recommend not to specify both thing template and data block in the request payload.

- If you specify both thing template and data block in the request payload, property values specified in the data block take the precedence.
- For example, the thing template contains properties and values for the property set type PST1. The data block contains properties and values for the property set type PST2. In such a scenario, the service creates a thing and assigns values from thing template and data block for the respective properties. If both thing template and data block contain properties and values for property set type PST1, the service creates a thing and assigns values from data block for the respective properties, and ignores the properties and values provided in the thing template.



The following table illustrates the combination of thing template and data block in the same request payload and the thing created as a result:

Thing Type	Thing Template	Data Block	Thing
TT1	TT1temp	Property set: npst1	T1
Property set: npst1	Property set: npst1	Property P1: 200	Property P1: 200
Property set type: PST1	Property P1: 100	Property P2: XYZ	Property P2: XYZ
Properties: P1, P2	Property P2: ABC	Property set: npst2	Property P3: 2017
Property set: npst2	Property set: npst2	Property P3: 2017	Property P4: 2.8
Property set type: PST2	Property P3: 2018	Property P4: 2.8	
Properties: P3, P4	Property P4: 1.2		
	TT1temp	Property set: npst2	T1
	Property set: npst1	Property P3: 2017	Property P1: 100
	Property P1: 100	Property P4: 2.8	Property P2: ABC
	Property P2: ABC		Property P3: 2017
			Property P4: 2.8

#### Note

- Thing type and thing have a one-to-many relationship. That is to say, one thing type may be referenced by many things, while each thing is defined by exactly one thing type. Similarly, thing template and thing have a one-to-many relationship. You can create multiple things based on a thing template, while each thing is created based on exactly one thing template.
- The thing type name and the thing template name must be fully qualified names that are prefixed with the package name. For example, `core.automobiles:ABCXSeries`.
- When you assign an object group to a thing, you are allowed to carry out the functions based on the capabilities defined for the user group. For more information about object groups, refer to [ObjectGroup \[page 206\]](#) and for more information about capabilities, refer to [Capabilities \[page 260\]](#).
- The data block in the request payload must not contain multiple data for the same property set.

## Thing to Sensors Assignment

### Prerequisites

Configure the IoT service credentials using Thing Modeler data mapping credentials API. See [Create Credentials \[page 754\]](#).

To onboard Thing Type along with its mappings with Sensor Types through convenience services, see [Thing Type and Sensor Type Mapping \[page 577\]](#).

### Functional Overview

Thing REST service supports auto creation of Device, Sensors, Thing and Assignment in a single API without the need of separate calls to IoT Service APIs.

This will be achieved by the below implementation approach:

- REST Payload of Thing API now includes device, sensors and assignment object structure.
- This API will internally perform the following actions to create the artifacts:
  - REST APIs of IoT Services create Devices and Sensors and obtain the object IDs.
  - Creates a Thing.
  - Assigns sensors to the Thing.

### Association between thing and custom master data

You can associate a thing instance to one or more object instances while creating the thing. This is possible only if an association is already created between the thing type and the object type, and one or more object instances are already created for the object types.

For example, the thing type **Truck** has an association with the object type **Vehicle** and the relationship is **HAS A**. You have already created one or object instances for the object type **Vehicle**. Based on this thing type and object type association, you can associate a thing instance of thing type **Truck** with one or more object instances of object type **Vehicle** while creating the thing instance. You can associate a thing with multiple object instances in a single POST request call.

#### i Note

- You can associate a thing with multiple objects of a specific object type, only if the association type is *onetomany* or *manytomany* between the thing type and the object type.
- You can associate a thing with multiple objects of different object types.
- You cannot add association details or update the association details when you update a thing.
- You cannot associate a thing instance with one or more thing instances using this service. It is always the association between the thing instance and the object instances.
- The expiry time of the thing and object association must be greater than the current date and less than the expiry time defined for the corresponding object type and thing type association.

#### Association type between thing type

TT1 and object type OT1

(OT1 has object instances obj1, obj2, obj3)

	Things of thing type TT1	Can be associated
onetoone	thing1	only with obj1 or obj2 or obj3
onetomany	thing1	with obj1 and obj2
manytoone	thing1	with obj1
	thing2	with obj1
manytomany		All thing instances of TT1 can be associated with one or more object instances of OT1

For more information, see [Custom Master Data \[page 598\]](#).

## Naming Convention

Only the following characters are supported for `_externalID` and `_name`:

- Uppercase and lowercase alphabets *a* through *z* and *A* through *Z*
- Numeric digits *0* through *9*
- Punctuation marks hyphen (-), underscore (\_), colon (:), and full-stop (.)

Only the following characters are supported for `_alternateID`:

- Uppercase and lowercase alphabets *a* through *z* and *A* through *Z*
- Numeric digits *0* through *9*
- Punctuation marks hyphen (-), underscore (\_)

## Request

**URI:** `/Things`

**HTTP Method:** *POST*

**Permissions:**

`<thing>.c`

## Request Properties

### Thing Object Properties

Property	Type	Mandatory	Maximum Length	Description
<code>_externalID</code>	string	No	255	Thing identifier (which is not unique) from an external system. You can create multiple things with the same external ID.  For example, <code>ABCXSeries001</code> is an external ID of a thing generated by an external system.
<code>_alternateID</code>	string	No	32	Thing identifier which is unique only within a specific tenant. You cannot change the alternate thing ID.
<code>_name</code>	string	Yes	255	Descriptive name of the thing.  For example, <code>ABCXSeries</code> is a descriptive name used to refer to a thing.
<code>_description</code>	string	Yes	60	Description of a Thing in the specified language as defined in <a href="#">ISO Language Codes [page 1242]</a> . The ISO code for the language is used here.
<code>_thingType</code>	Array of thing types	No	81	Array of thing types for a thing. It must correspond to one of the thing types found in a package. Currently, only one element is supported.

Property	Type	Mandatory	Maximum Length	Description
_thingTemplate	Array of thing templates	No		Array of thing templates for a thing. It must correspond to one of the thing types found in a package. Currently, only one element is supported.
_objectGroup	string	Yes	32	Object group ID for which the Thing has to be assigned to.
_location	string	No	32	Identifier for a Thing Location.
				<b>i Note</b> Assigning a value for this property is not mandatory. However, if a value is provided, it must be the unique identifier of the thing location (locationID).
_customer	string	No	32	Identifier for a Business Partner.
				<b>i Note</b> Assigning a value for this property is not mandatory. However, if a value is provided, it must be the unique identifier of the business partner (businessPartnerID of the relevant role).

### Thing Data Object Properties

Property	Type	Description
_time	Date or timestamp	Date or time stamp for the properties. It appears only if relevant.
{property type name}	property type	Property type name as found in the package

### Sensor Assignment Properties

Parent Key	Property	Type	Mandatory	Maximum Length	Description
_assignment	_mappingId	GUID	Yes	32	Thing Type to Sensor Type Mapping ID
_devices	_name	string	Yes	81	Name of the IoT Service device

Parent Key	Property	Type	Mandatory	Maximum Length	Description
_devices	_gatewayName	string	Yes, if device needs to be newly created	81	Gateway ID of IoT Service  This is immutable and cannot be updated for an existing device. It means if the payload has a device with an ID, then that device should not have the gateway name.
_sensors	_name	string	Yes	81	Name of the IoT Service sensor
_sensors	_sensorTypeId	string	Yes	81	Sensor Type ID of the IoT Service sensor

### Custom Master Data Association Object Properties

Property	Type	Mandatory	Maximum Length	Description
AssociatedObjectID	GUID	Yes	32	Unique identifier of the object instance created for an object type
AssociatedEntity	string	Yes	81	Unique name of the thing type
Relationship	string	Yes	50	Type of association between the thing type and the associated object type. The only supported values are <b>HAS A</b> , <b>IS A</b> , and <b>PART OF</b> .
StartTime	Timestamp	No	50	Time when the association between the thing instances and object instances is created
ExpiryTime	Timestamp	No	50	Expiry time after which the association between the thing instances and object instances becomes invalid

### Request Example

```
/Things
```

### Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the structure as shown in the following examples:

- [Example 1 – Create a thing for a specific thing type \[page 722\]](#)
- [Example 2 – Create a thing with a specific alternate thing ID \[page 723\]](#)
- [Example 3 – Create a thing based on a thing template \[page 723\]](#)
- [Example 4 – Create a thing along with data for properties \[page 724\]](#)
- [Example 5 – Create a thing along with the assignment to sensors \[page 725\]](#)
- [Example 6 – Create thing with custom master data association \[page 725\]](#)

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
201	Thing created successfully

#### i Note

The URL of the thing created is returned in the **Location Header** of the response.

## Related Information

[ObjectGroup \[page 206\]](#)

[Location \[page 133\]](#)

[Business Partner \[page 25\]](#)

### 12.3.1.1 Example 1 – Create a thing for a specific thing type

The following request payload creates a thing of thing type `core.automobiles:ABCXSeries`.

```
{
  "_externalId": "ABCXSeries001",
  "_name": "ABCXSeries",
  "_description": {
    "_en": "Car Model ABC  xSeries 001"
  },
  "_thingType": [
    "core.automobiles:ABCXSeries"
  ],
  "_location": "BD622E3DA6A14C4D87CDEC935B000001",
  "_customer": "4D622E3DA6A14C4D87CDEC935B000001",
  "_objectGroup": "473BB1AFC4A4415791543C3F2CFDD295"
```

```
}
```

#### **i Note**

An error occurs if the `_objectGroup` property is not specified, empty, or has an invalid ID assigned to it.

## **Related Information**

[Create a Thing \[page 716\]](#)

### **12.3.1.2 Example 2 – Create a thing with a specific alternate thing ID**

The following request payload creates a thing with alternate thing ID.

```
{
  "_externalId": "ABCXSeries001",
  "_alternateId": " ABCXSeries001-alternateId",
  "_name": "ABCXSeries",
  "_description": {
    "en": "Car Model ABC  xSeries 001"
  },
  "_thingType": [
    "core.automobiles:ABCXSeries"
  ],
  "_location": "BD622E3DA6A14C4D87CDEC935B000001",
  "_customer": "4D622E3DA6A14C4D87CDEC935B000001",
  "_objectGroup": "473BB1AFC4A4415791543C3F2CFDD295"
}
```

## **Related Information**

[Create a Thing \[page 716\]](#)

### **12.3.1.3 Example 3 – Create a thing based on a thing template**

The following request payload creates a thing based on the a thing template `core.automobiles:ABC2SeriesTemplate`.

```
{
  "_externalId": "ABCXSeries001",
  "_alternateId": " ABCXSeries001-alternateId",
  "_name": "ABCXSeries",
```

```

    "_description": {
      "en": "Car Model ABC  xSeries 001"
    },
    "_thingTemplate": [
      "core.automobiles:ABC2SeriesTemplate"
    ],
    "_location": "BD622E3DA6A14C4D87CDEC935B000001",
    "_customer": "4D622E3DA6A14C4D87CDEC935B000001",
    "_objectGroup": "473BB1AFC4A4415791543C3F2CFDD295"
  }
}

```

## Related Information

[Create a Thing \[page 716\]](#)

### 12.3.1.4 Example 4 – Create a thing along with data for properties

The following request payload creates a thing along with data for properties of the property sets. The data block in this example contains values for properties `BodyStyle`, `Class`, and `WheelBase` of property set `car` and for properties `OriginPlace` and `Country` of property set `location`.

```

{
  "_externalId": "ABCXSeries001",
  "_name": "ABCXSeries",
  "_description": {
    "en": "Car Model ABC  xSeries 001"
  },
  "_thingType": [
    "core.automobiles:ABCXSeries"
  ],
  "_objectGroup": "473BB1AFC4A4415791543C3F2CFDD295",
  "data": {
    "car": {
      "BodyStyle": "Sedan",
      "Class": "E",
      "WheelBase": "invalid"
    },
    "location": {
      "OriginPlace": "Munich",
      "Country": "Germany"
    }
  }
}

```

## Related Information

[Create a Thing \[page 716\]](#)



### 12.3.1.5 Example 5 – Create a thing along with the assignment to sensors

Thing REST service supports auto creation of Device, Sensors, Thing and Assignment in a single API without the need of separate calls to IoT Service APIs. For more information, see [Create a Thing \[page 716\]](#)

The following request payload creates a thing along with its sensor assignment:

```
{
  "_externalId": "Demo-Cooler-Thing-01",
  "_name": "Demo-Cooler-Thing-01",
  "description": {
    "en": "Demo Cooler"
  },
  "thingType": [
    "core.coolerpkg:CoolerType"
  ],
  "objectGroup": "C18E47BEEF1D41149BB3AFFEB38BF66B",
  "assignment": {
    "_devices": [
      {
        "_name": "DemoCooler",
        "gatewayName": "3",
        "sensors": [
          {
            "_name": "ConvPressureSensor",
            "_sensorTypeId": "5e70f20b-c0cb-4146-8c8b-eeae96a86f01"
          },
          {
            "_name": "ConvReedSensor",
            "_sensorTypeId": "0b0aa063-3ddc-4920-8a8a-50744b4676d5"
          },
          {
            "_name": "ConvCoolerSensor",
            "_sensorTypeId": "3376ab0a-34b2-4c46-877b-3d76f676d77f"
          }
        ]
      }
    ]
  },
  "_mappingId": "b807fece-b5b9-44ed-a7b4-4daffba5a691"
}
```

#### Related Information

[Create a Thing \[page 716\]](#)

### 12.3.1.6 Example 6 – Create thing with custom master data association

The following request payload creates a thing along with custom master data association.

```
{
```

```

    "_externalId": " ABCXSeries001",
    "_name": " ABCXSeries",
    "_description": {
      "en": " Car Model ABC xSeries 001"
    },
    "_thingType": [
      " core.automobiles:ABCXSeries"
    ],
    "_objectGroup": " 473BB1AFC4A4415791543C3F2CFDD295",
    "customDataAssociation": [
      {
        "AssociatedObjectID": "FrontRWheel",
        "AssociatedEntity": "core.automobiles:Wheel",
        "Relationship": "HAS A",
        "Descriptions": [
          {
            "LanguageCode": "en",
            "Description": "Association between Thing ABCXSeries and
FrontRWheel"
          }
        ],
        "StartTime": "2018-05-17T13:00:56.229Z",
        "ExpiryTime": "2018-06-17T12:00:56.229Z"
      },
      {
        "AssociatedObjectID": "RearRWheel",
        "AssociatedEntity": "core.automobiles:Wheel",
        "Relationship": "HAS A",
        "StartTime": "2018-05-17T13:00:56.229Z",
        "ExpiryTime": "2018-06-17T12:00:56.229Z"
      },
      {
        "AssociatedObjectID": "RearLWheel",
        "AssociatedEntity": "core.automobiles:Wheel",
        "Relationship": "HAS A",
        "Descriptions": [
          {
            "LanguageCode": "en",
            "Description": " Association between Thing ABCXSeries and
RearLWheel"
          }
        ],
        "StartTime": "2018-05-17T13:00:56.229Z",
        "ExpiryTime": "2018-06-17T12:00:56.229Z"
      }
    ]
  }
}

```

From the given request payload, you can infer the following:

- Thing type `core.automobiles:ABCXSeries` is already associated with the object type `core.automobiles:Wheel`. The association type is *onetomany* and the relationship is **HAS A**.
- Object instances `FrontRWheel` and `RearRWheel` are already created for the object type `core.automobiles:Wheel`. Note that you must provide the auto generated GUID for the object instances.
- The new thing instance is created and associated with the object instances `FrontRWheel` and `RearRWheel`.

## Related Information

[Create a Thing \[page 716\]](#)

## 12.3.2 Create Multiple Things

With this method, you can create multiple thing instances for a specific thing type.

When you create multiple things using this method, it is mandatory that you provide a unique alternate ID for each and every thing instance. The method validates the request payload and initiates the thing creation job at the backend. In addition to the thing type, you can specify the thing template created for that thing type.

As a result, the method displays the response message 202 accepted and generates a correlation ID. You can find the correlation ID in the response header. For every correlation ID, the method generates unique log IDs one each for the number of things specified in the request payload. Using the correlation ID and the log ID, you can know the success or failure status of individual things. For more information, see [Monitoring and Logging \[page 1205\]](#).

The multiple things that you are planning to create using this method, must belong a specific thing type. You cannot create things for multiple thing types or multiple thing templates in the same request payload. The things in the request payload must belong to only one object group and you cannot provide different object groups for different things.

### i Note

Only creation of multiple things is supported, and you cannot update multiple things.

**Request URI:** /v2/Things

### Response Header

Header	Description
correlationid	Unique correlation ID generated for the service executed.

### Payload Examples

- [Example 1 – Create multiple things for a thing type \[page 727\]](#)
- [Example 2 – Create multiple things using thing template \[page 728\]](#)

### 12.3.2.1 Example 1 – Create multiple things for a thing type

The following request payload creates multiple things for the specified thing type:

```
{
  "thingType": "core.automobiles:ABCXSeries",
  "_objectGroup": "EFE8B81D89194CFDBA936722EE27053B",
  "things": [
    {
      "_externalId": "ABCXCar-extid",
      "_alternateId": "ABCXCar-altid",
      "_name": "ABCXCar",
      "_description": {
        "_en": "ABC Car of X Series"
      },
      "_location": "029611A5D3B64815B4DA404B72DDD16F",
      "_customer": "3C4E1C7A1E984309A608623C2425DA0D"
    },
    {

```

```

    "_externalId": "ABCX1Car-extid",
    "_alternateId": " ABCX1Car-altid",
    "_name": "ABCX1Car",
    "_description": {
      "_en": "ABC Car of X1 Series"
    },
    "_location": "029611A5D3B64815B4DA404B72DDD16F",
    "_customer": "3C4E1C7A1E984309A608623C2425DA0D"
  },
  {
    "_externalId": "ABCX2Car-extid",
    "_alternateId": "ABCX2Car-altid",
    "_name": "ABCX2Car",
    "_description": {
      "_en": "ABC Car of X2 Series"
    },
    "_location": "029611A5D3B64815B4DA404B72DDD16F",
    "_customer": "3C4E1C7A1E984309A608623C2425DA0D"
  }
]
}

```

## Related Information

[Create Multiple Things \[page 727\]](#)

### 12.3.2.2 Example 2 – Create multiple things using thing template

The following request payload creates multiple things for the specified thing type. In addition, based on the thing template specified, property data will be created for the thing.

```

{
  "_thingType": "core.automobiles:ABCXSeries",
  "_thingTemplate": "core.automobiles:ABCXSeriesTemp",
  "_objectGroup": "EFE8B81D89194CFDBA936722EE27053B",
  "_things": [
    {
      "_externalId": "ABCXCar-extid",
      "_alternateId": "ABCXCar-altid",
      "_name": "ABCXCar",
      "_description": {
        "_en": "ABC Car of X Series"
      },
      "_location": "029611A5D3B64815B4DA404B72DDD16F",
      "_customer": "3C4E1C7A1E984309A608623C2425DA0D"
    },
    {
      "_externalId": "ABCX1Car-extid",
      "_alternateId": " ABCX1Car-altid",
      "_name": "ABCX1Car",
      "_description": {
        "_en": "ABC Car of X1 Series"
      },
      "_location": "029611A5D3B64815B4DA404B72DDD16F",
      "_customer": "3C4E1C7A1E984309A608623C2425DA0D"
    }
  ]
}

```

```

{
  "_externalId": "ABCX2Car-extid",
  "_alternateId": "ABCX2Car-altid",
  "_name": "ABCX2Car",
  "_description": {
    "_en": "ABC Car of X2 Series"
  },
  "_location": "029611A5D3B64815B4DA404B72DDD16F",
  "_customer": "3C4E1C7A1E984309A608623C2425DA0D"
}
]
}

```

## Related Information

[Create Multiple Things \[page 727\]](#)

### 12.3.3 Read a Thing by Alternate ID

With this method, you can retrieve details of a thing with a specific alternate thing ID.

#### Request

**URI:** `/ThingsByAlternateId('<alternateId>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thing>.r`

#### Request Headers

Parameter	Required	Description
Accept-Language	No	Language of the thing description. The default language is EN.

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header CONTENT-ENCODING with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Parameters

Parameter	Required	Data Type	Description
alternate ID	Yes	String	Unique thing identifier assigned by the user while creating a thing.

## Request Example

```
/ThingsByAlternateId('ABCXSeries001-alternateId')
```

## Response

### Response Properties

Property	Type	Description
_id	String	Unique thing identifier generated by the system while creating a thing.
_externalId	String	Thing identifier from an external system.
_alternateId	String	Thing identifier which is unique only within a specific tenant.
_name	String	Name of the thing.
_description	String	Description of a thing in a specified language.
_thingType	Array of thing types	Array of thing types for the thing.
_location	String	Identifier for the location.
_customer	String	Identifier for the business partner.
_objectGroup	String	Object group ID for which the Thing has to be assigned.

## Response Status and Error Codes

Code	Description
200	Thing with specific alternate ID retrieved successfully.

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

```
{
  "id": "503B8711B5D0445688E7FF7ACFC2177A",
  "_externalId": "ABCXSeries001",
  "_alternateId": "ABCXSeries001-alternateId",
  "_name": "ABCXSeries",
  "_description": {
    "en": "Car Model ABC   xSeries 001"
  },
  "_thingType": [
    "core.automobiles:ABCXSeries"
  ],
  "location": "BD622E3DA6A14C4D87CDEC935B000001",
  "customer": "4D622E3DA6A14C4D87CDEC935B000001",
  "_objectGroup": "473BB1AFC4A4415791543C3F2CFDD295"
}
```

## 12.3.4 Read a Thing

With this method, you can read the generic fields and the thing type associated with the specified thing.

### i Note

If you have created a thing using a thing template with the file property specified, this method does not retrieve the file content for the thing. You must use /

Snapshot(thingId='<ID>', fromTime='<timestamp>', dataCategory='<dataCategory>')

or /v2/

Snapshot(thingId='<ID>', fromTime='<timestamp>', toTime='<timestamp>', dataCategory='<dataCategory>') to retrieve the file content of a thing.

## Request

URI: /Things ('<ID>')

HTTP Method: *GET*

## Permissions:

<thing>.r

## Request Header Parameters

Parameter	Required	Description
Accept-Language	No	Language of the thing description. The default language is <a href="#">EN</a> .
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header <code>CONTENT-ENCODING</code> with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Parameter

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>

## Request Example

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')
```

The below request will get the thing and its assignment details.

```
{
  "_id": "92B98497566F48E4A0867569941F9CF1",
  "_externalId": "Demo-Cooler-Thing-01",
  "_name": "Demo-Cooler-Thing-01",
  "_description": {
    "_en": "Demo Cooler"
  },
  "_thingType": [
    "iotae.sandboxdibd1.conv.services.demo.package:CoolerType"
  ],
  "CreatedByUser": "testuser",
  "CreatedTime": "2019-02-07T06:10:41.899Z",
  "LastChangedByUser": "testuser",
  "LastChangedTime": "2019-02-07T06:10:41.899Z",
  "_assignment": {
    "_id": "79721DA29EAA41A6ACACEC7920F9A9D5",
    "_devices": [
      {
        "_id": "174",
        "_name": "DemoCooler",
        "_gatewayName": "3",
        "_sensors": [
          {
            "_id": "733",
            "_name": "ConvPressureSensor",
            "_sensorTypeId": "5e70f20b-c0cb-4146-8c8b-eeae96a86f01",
```



```

        "_operation": null
      },
      {
        "_id": "734",
        "_name": "ConvReedSensor",
        "_sensorTypeId": "0b0aa063-3ddc-4920-8a8a-50744b4676d5",
        "_operation": null
      },
      {
        "_id": "732",
        "_name": "ConvCoolerSensor",
        "_sensorTypeId": "3376ab0a-34b2-4c46-877b-3d76f676d77f",
        "_operation": null
      }
    ]
  },
  "_mappingId": "b807fece-b5b9-44ed-a7b4-4daffba5a691"
},
"_objectGroup": "C18E47BEEF1D41149BB3AFFEB38BF66B"
}

```

## Response

Format: *JSON*

### Response Properties

Property	Type	Description
_id	string	Unique thing identifier generated by the system while creating a thing.
_externalId	string	Thing identifier from an external system.
_alternateId	string	Thing identifier which is unique only within a specific tenant.
_name	string	Name of the thing.
_description	string	Description of a thing in a specified language.
_thingType	Array of thing types	Array of thing types for the thing.
_location	string	Identifier for the location.
_customer	string	Identifier for the business partner.
_objectGroup	string	Object group ID for which the Thing has to be assigned to.

### Response Status and Error Codes

Code	Description
200	Thing retrieved successfully

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  "_id": "503B8711B5D0445688E7FF7ACFC2177A",
  "_externalId": "ABCXSeries001",
  "_name": "ABCXSeries",
  "_description": {
    "en": "Car Model ABC   xSeries 001"
  },
  "_thingType": [
    "core.automobiles:ABCXSeries"
  ],
  "location": "BD622E3DA6A14C4D87CDEC935B000001",
  "_customer": "4D622E3DA6A14C4D87CDEC935B000001",
  "_objectGroup": "473BB1AFC4A4415791543C3F2CFDD295",
  "CreatedByUser": "abc@xyz.com",
  "CreatedTime": "2018-09-04T11:42:14.662Z",
  "LastChangedByUser": "abc@xyz.com",
  "LastChangedTime": "2018-09-04T11:42:14.662Z"
}
```

If you assigned an alternate thing ID while creating a thing, the response payload contains the `_alternateId` field.

```
{
  "_id": "503B8711B5D0445688E7FF7ACFC2177A",
  "_externalId": "ABCXSeries001",
  "_alternateId": "ABCXSeries001-alternateId",
  "_name": "ABCXSeries",
  "_description": {
    "en": "Car Model ABC   xSeries 001"
  },
  "_thingType": [
    "core.automobiles:ABCXSeries"
  ],
  "location": "BD622E3DA6A14C4D87CDEC935B000001",
  "_customer": "4D622E3DA6A14C4D87CDEC935B000001",
  "_objectGroup": "473BB1AFC4A4415791543C3F2CFDD295",
  "CreatedByUser": "abc@xyz.com",
  "CreatedTime": "2018-09-04T11:42:14.662Z",
  "LastChangedByUser": "abc@xyz.com",
  "LastChangedTime": "2018-09-04T11:42:14.662Z"
}
```

## 12.3.5 Read Thing Configuration Details for a Thing

Retrieves metadata for a thing

With this method, you can retrieve thing configuration details for a specified thing. The result set contains thing types and property sets to which the specified thing belongs.

### Request

**URI:** `/Things('<thing ID')/Configuration`

**HTTP Method:** *GET*

Permissions: <thing>.r

## Request Headers

Header	Required	Values
Accept-Language	No	Language of the thing type description. The default language is <a href="#">EN</a> .
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header <code>CONTENT-ENCODING</code> with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the un-compressed response.

## Request Parameters

Parameter	Required	Data Type	Description
id	Yes	string	Internal identifier for a thing.

## Request Example

```
/Things('<thing ID>')/Configuration
```

## Response

Format: [JSON](#)

## Response Properties

Property	Type	Description
Value	Array of thing type objects	Array of thing type objects returned.
propertySets	Array of property set objects	Array of property set objects (that belong to the property set type) returned.

## Response Status and Error Codes

Code	Description
200	Configuration details retrieved successfully

## Payload

Format: [JSON](#)

Only media type [JSON](#) is supported. The [JSON](#) for this method has the following structure:

```
{
  "thingTypes": [{
    "name": "core.automobiles:ABC2Series",
    "description": {
      "en": "ABC2Series electric car"
    },
    "propertyTypes": [{
      "id": "engine",
      "description": {
        "en": "Engine Usage"
      },
      "propertySetType": "core.automobiles:Engine"
    }, {
      "id": "drive_fr",
      "description": {
        "en": "Drive Front Right"
      },
      "propertySetType": "core.automobiles:DriveUnit"
    }, {
      "id": "drive_fl",
      "description": {
        "en": "Drive Front Left"
      },
      "propertySetType": "core.automobiles:DriveUnit"
    }, {
      "id": "car",
      "description": {
        "en": "Car Usage"
      },
      "propertySetType": "core.automobiles:Car"
    }, {
      "id": "wheel_fr",
      "description": {
        "en": "Wheel Front Right"
      },
      "propertySetType": "core.automobiles:Wheel"
    }, {
      "id": "wheel_fl",
      "description": {
        "en": "Wheel Front Left"
      },
      "propertySetType": "core.automobiles:Wheel"
    }, {
      "id": "wheel_rr",
      "description": {
        "en": "Wheel Rear Right"
      },
      "propertySetType": "core.automobiles:Wheel"
    }, {
      "id": "wheel_rl",
      "description": {
        "en": "Wheel Rear Left"
      },
      "propertySetType": "core.automobiles:Wheel"
    }
  ]
}, {
  "propertySetTypes": [{
    "name": "core.automobiles:Car",
    "description": {
      "en": "Master Data for car"
    },
    "dataCategory": "MasterData",
    "propertyTypes": [{
      "value": {
        "id": "BodyStyle",
        "description": {
```

```

        "en": "Car Body Style"
    },
    "type": "String",
    "length": "127"
  }, {
    "value": {
      "id": "Class",
      "description": {
        "en": "Car Class"
      },
      "type": "String",
      "length": "127"
    }
  }, {
    "value": {
      "id": "WheelBase",
      "description": {
        "en": "Car Wheel Base"
      },
      "type": "Numeric",
      "length": ""
    }
  }, {
    "value": {
      "id": "RegistrationNumber",
      "description": {
        "en": "Car Registration Number"
      },
      "type": "String",
      "length": "127"
    }
  }, {
    "value": {
      "id": "Colour",
      "description": {
        "en": "Car Colour"
      },
      "type": "String",
      "length": "127"
    }
  }
  ]]
}, {
  "name": "core.automobiles:Wheel",
  "description": {
    "en": "Sensory parameters for wheel"
  },
  "dataCategory": "TimeSeriesData",
  "propertyTypes": [{
    "value": {
      "id": "Pressure",
      "description": {
        "en": "Wheel pressure"
      },
      "type": "Numeric",
      "length": "",
      "unitOfMeasure": "PA"
    }
  }, {
    "value": {
      "id": "Temperature",
      "description": {
        "en": "Wheel temperature"
      },
      "type": "Numeric",
      "length": ""
    }
  }
  ]]
}

```

```

}, {
  "name": "core.automobiles:DriveUnit",
  "description": {
    "en": "Sensory parameters for drive unit"
  },
  "dataCategory": "TimeSeriesData",
  "propertyTypes": [{
    "value": {
      "id": "RotationSpeed",
      "description": {
        "en": "Drive unit rotation speed"
      },
      "type": "Numeric",
      "length": ""
    }
  }, {
    "value": {
      "id": "PowerConsumption",
      "description": {
        "en": "Drive unit power consumption"
      },
      "type": "Numeric",
      "length": ""
    }
  }
  ]
}, {
  "name": "core.automobiles:Engine",
  "description": {
    "en": "Sensory parameters for engine"
  },
  "dataCategory": "MasterData",
  "propertyTypes": [{
    "value": {
      "id": "Type",
      "description": {
        "en": "Engine type"
      },
      "type": "String",
      "length": "127"
    }
  }, {
    "value": {
      "id": "Capacity",
      "description": {
        "en": "Engine capacity"
      },
      "type": "Numeric",
      "length": "",
      "unitOfMeasure": "MM"
    }
  }, {
    "value": {
      "id": "Torque",
      "description": {
        "en": "Engine torque"
      },
      "type": "Numeric",
      "length": "",
      "unitOfMeasure": "NM"
    }
  }, {
    "value": {
      "id": "Transmission",
      "description": {
        "en": "Engine transmission"
      },
      "type": "String",
      "length": "254",

```

```

        "unitOfMeasure": ""
    }
}
}

```

## Related Information

[Thing Configuration \[page 352\]](#)

### 12.3.6 Update a Thing

With this method, you can update attributes such as external ID, name, or object group of a thing.

In the payload of the request, you assign values to those fields that you want to modify. For example, to set a new name, you may pass the desired string for the `<_name>` field in the payload of the service request.

#### i Note

`<_externalID>` and `<_customer>` are optional fields in the request payload and this method updates these optional fields as follows:

- If you specify a value for these optional fields in the request payload, the method replaces the old value with the new value.
- If you do not specify these fields in the request payload, the method retains the original value.

## Request

**URI:** `/Things ('<ID>')`

**HTTP Method:** *POST*

**Permissions:**

`<thing>.c`

#### Request Parameter

Parameter	Required	Data Type	Description
ID	Yes	string	<p>You can specify one of the following values:</p> <ul style="list-style-type: none"> <li>• Unique thing identifier generated by the system while creating a thing.</li> <li>• Alternate thing identifier which is unique only within a specific tenant.</li> </ul>

## Request Example

/Things ('503B8711B5D0445688E7FF7ACFC2177A')

## Sensor Assignment Properties

Parent Key	Property	Type	Mandatory	Maximum Length	Description
_assignment	_id	GUID	Yes	32	Unique identifier of the assignment
_assignment	_mappingId	GUID	Yes	32	Thing Type to Sensor Type Mapping ID
_devices	_id	string	Yes (if sensors are to be added, or updated or deleted to/from the device, and also if the device name is to be updated)	81	ID of the IoT Service device
_devices	_name	string	Yes	81	Name of the IoT Service device
_devices	_gateway-Name	string	Yes, if device needs to be newly created	81	Gateway ID of IoT Service  This is immutable and cannot be updated for an existing device. It means if the payload has a device with an ID, then that device should not have the gateway name.
_sensors	_id	string	Yes (if sensor is to be updated or deleted)	81	ID of the IoT Service sensor
_sensors	_operation	string	Yes (if sensor is to be updated or deleted)		ADD: Creates a new sensor, adds it to the device in IoT Services, and adds the sensor to the assignment in IoT AE.  DELETE: Deletes the existing sensor from IoT Services, and deletes the sensor from the assignment in IoT AE.  UPDATE: Updates the sensor name or sensor type ID of the sensor.

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
```



```

    "_externalId": "ABCXSeries001",
    "_name": "ABCXSeries",
    "_description": {
      "en": "Car Model ABC   xSeries 001"
    },
    "_thingType": [
      "core.automobiles:ABCXSeries"
    ],
    "_location": "BD622E3DA6A14C4D87CDEC935B000001",
    "_customer": "4D622E3DA6A14C4D87CDEC935B000001",
    "_objectGroup": "473BB1AFC4A4415791543C3F2CFDD295"
  }
}

```

### Note

- An error occurs if the **\_objectGroup** property is not specified, empty, or has an invalid ID assigned to it.
- You cannot modify the thing type when you update a thing.

The below request deletes the sensor that has an ID "733", updates the name of the sensor that has an ID "734" and adds a new sensor NewConvPressureSensor to the assignment.

[https://apiiot-mds.cfapps.sap.hana.ondemand.com/Things\('92B98497566F48E4A0867569941F9CF1'\)](https://apiiot-mds.cfapps.sap.hana.ondemand.com/Things('92B98497566F48E4A0867569941F9CF1'))

```

{
  "_id": "92B98497566F48E4A0867569941F9CF1",
  "_externalId": "Demo-Cooler-Thing-01",
  "_name": "Demo-Cooler-Thing-01",
  "_description": {
    "en": "Demo Cooler"
  },
  "_thingType": [
    "iotae.sandboxhdibdl.conv.services.demo.package:CoolerType"
  ],
  "_assignment": {
    "_id": "79721DA29EAA41A6ACACEC7920F9A9D5",
    "_devices": [
      {
        "_id": "174",
        "_name": "DemoCooler",
        "_sensors": [
          {
            "_id": "733",
            "_name": "ConvPressureSensor",
            "_sensorTypeId": "5e70f20b-c0cb-4146-8c8b-eeae96a86f01",
            "_operation": "DELETE"
          },
          {
            "_id": "734",
            "_name": "ConvReedSensorUpdated",
            "_sensorTypeId": "0b0aa063-3ddc-4920-8a8a-50744b4676d5",
            "_operation": "UPDATE"
          },
          {
            "_name": "NewConvPressureSensor",
            "_sensorTypeId": "3376ab0a-34b2-4c46-877b-3d76f676d77f",
            "_operation": "ADD"
          }
        ]
      }
    ]
  },
  "_mappingId": "b807fece-b5b9-44ed-a7b4-4daffba5a691"
},
"_objectGroup": "C18E47BEEF1D41149BB3AFFEB38BF66B"
}

```

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
200	Thing updated successfully

## 12.3.7 Delete a Thing

This service is used to delete the specified thing.

If a thing is deleted, all the corresponding time series data, events, reference properties, files, sensor assignment, and custom master data association are also deleted.

- If the thing is associated with a thing hierarchy (parent-child relationship created between things), the system displays an error message and does not delete the thing. Start removing the parent-child relationship between the things in the corresponding hierarchies, and then delete the thing.
- If the thing is part of a thing group, the system displays an error message and does not delete the thing.

### i Note

Files for a thing are deleted only if the user has the `<file>.da` scope authorization assigned.

## Request

URI: `/Things ('<ID>')`

HTTP Method: *DELETE*

Permissions:

`<thing>.d`

`<file>.da`

## Request Parameter

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>

## Request Example

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')
```

```
/Things('thing ID')
```

## Response

Format: [JSON](#)

## Response Status and Error Codes

Code	Description
204	Thing deleted successfully

## 12.3.8 Read all Things

With this method, you can read all the available things.

A user who is assigned to a tenant for a cloud platform provider (TO) with `crossTenantScope` Capability can read things that are created in all tenants, regardless of the tenant to which they are logged on. For more information about different capabilities, see [Capabilities \[page 260\]](#).

### i Note

Retrieving number of objects more than the recommended limits with only one collective GET request can lead to high system load and a significant increase of response times.

To avoid performance decline, the system in future versions will automatically limit collective GET requests to a maximum of 500 objects that can be retrieved with one single call.

Use of `$count` as well degrades performance. Hence, a use of `$count` must also be judiciously chosen.

Therefore, for use cases where it is crucial for you to ensure that objects more than 500 have to be retrieved, you need to use paging options such as `$top` and `$skip`.

## Request

URI: /Things

HTTP Method: [GET](#)

Permissions:

<thing>.r

### Request Header Parameters

Parameter	Required	Description
Accept-Language	No	Language of the Thing description. The default language is <a href="#">en</a> .
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header <code>CONTENT-ENCODING</code> with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

### Request Parameters

None

### Query String Parameters

Parameter	Required	Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$count	No	Boolean	<p>Total number of thing objects available based on the current user's authorizations; permissible values are <a href="#">true</a> and <a href="#">false</a>.</p> <p>The following results apply, if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"><li>• If you use this with the \$top and \$skip parameters, the result contains the total count of thing objects available.</li><li>• If you use this with \$filter parameter, the result contains the total count of thing objects based on the defined filter condition.</li></ul>

Parameter	Required	Type	Description
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; valid fields are <a href="#">_id</a> , <a href="#">_name</a> , <a href="#">_externalId</a> .  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$filter	No	String	Filter condition to be applied; valid fields are <a href="#">_id</a> , <a href="#">_name</a> , <a href="#">_externalId</a> , <a href="#">_alternateId</a> , <a href="#">_thingType</a> , <a href="#">_objectGroup</a> . The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> .  Multiple filters are supported using the <a href="#">and</a> separator.  You can use the string functions such as <a href="#">substringof</a> , <a href="#">startswith</a> , and <a href="#">endswith</a> to search data using <code>\$filter</code> . Valid fields are <a href="#">_externalId</a> , <a href="#">_name</a> , <a href="#">_thingType</a> , and <a href="#">_alternateId</a> .
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are <a href="#">_id</a> , <a href="#">_name</a> , <a href="#">_externalId</a> , <a href="#">_thingType</a> , <a href="#">_location</a> , <a href="#">_customer</a> .

## Request Example

```
/Things?$filter=_thingType eq 'core.automobiles:ABCXSeries'&$count=true
```

Retrieves all things where the thing type is `core.automobiles:ABCXSeries` and the total number of thing objects that match the filter criterion.

```
/Things?$filter=_thingType eq 'core.automobiles:ABCXSeries'&$count=true&
$select=_thingType, _externalID
```

Retrieves all things where the thing type is `core.automobiles:ABCXSeries` and the total number of thing objects that match the filter criterion.. However, with `$select` query parameter, only the fields [\\_thingType](#) and [\\_externalId](#) are displayed for all the retrieved things.

```
/Things?$filter=_alternateId eq 'ABCXSeries001-alternateId'
```

Retrieves the thing with alternate thing ID `ABCXSeries001-alternateId`.

```
/Things?$filter=_alternateId eq 'ABCXSeries001-alternateId' or _alternateId eq
'ABC2Series-alternateId'
```

Retrieves two things with their respective alternate thing ID `ABCXSeries001-alternateId` and `ABC2Series-alternateId`.

## Response

Format: [JSON](#)

## Response Properties

Property	Type	Description
value	Array of thing objects	Array of things returned upon application of the filter conditions.

## Response Status and Error Codes

Code	Description
200	Things retrieved successfully

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  "count": 1,
  "value": [{
    "id": "<thingID>",
    "_externalId": "ABCXSeries001",
    "name": "ABCXSeries",
    "description": {
      "en": "Car Model ABC  xSeries 001"
    },
    "_thingType": [
      "core.automobiles:ABCXSeries"
    ],
    "location": "BD622E3DA6A14C4D87CDEC935B000001",
    "customer": "4D622E3DA6A14C4D87CDEC935B000001",
    "objectGroup": "473BB1AFC4A4415791543C3F2CFDD295",
    "assignmentId": "79721DA29EAA41A6ACACEC7920F9A9D5",
    "CreatedByUser": "abc@xyz.com",
    "CreatedTime": "2018-09-04T11:42:14.662Z",
    "LastChangedByUser": "abc@xyz.com",
    "LastChangedTime": "2018-09-04T11:42:14.662Z"
  }]
}
```

## 12.3.9 Payload of CapabilityType "Thing"

Presents the payload structure of CapabilityType "Thing"

The payload of the CapabilityType service consists of two main parts:

- **vocabulary:** This part mirrors the payload elements of the Thing service. Any values that you enter here limit the scope of objects to those things that match the passed values.
- **actions:** This part defines which of the three possible types of action (read, write delete) shall be allowed for the Thing objects specified in the vocabulary.

## Payload

```
{
  "ID": "FFFFFFFFF00000000FFFFFFFFF0003",
  "name": "thing:thing",
  "vocabulary": {
    "attributes": [
      {
        "path": {
          "field": "Customer",
          "dimension": "$property"
        },
        "valueType": {
          "type": "String",
          "length": 127
        }
      },
      {
        "path": {
          "field": "ExternalId",
          "dimension": "$thing"
        },
        "valueType": {
          "type": "String",
          "length": 127
        }
      },
      {
        "path": {
          "field": "Location",
          "dimension": "$property"
        },
        "valueType": {
          "type": "String",
          "length": 127
        }
      },
      {
        "path": {
          "field": "ThingType",
          "dimension": "$thingtypes"
        },
        "valueType": {
          "type": "String",
          "length": 127
        }
      },
      {
        "path": {
          "field": "ObjectGroup",
          "dimension": "$assignment"
        },
        "valueType": {
          "type": "HierarchyAssignment"
        }
      },
      {
        "path": {
          "field": "Thing",
          "dimension": "$thing"
        },
        "valueType": {
          "type": "String",
          "length": 127
        }
      }
    ]
  }
}
```

```

        {
            "path": {
                "field": "Name",
                "dimension": "$thing"
            },
            "valueType": {
                "type": "String",
                "length": 127
            }
        },
        {
            "path": {
                "field": "AlternateId",
                "dimension": "$thing"
            },
            "valueType": {
                "type": "String",
                "length": 127
            }
        }
    ],
    "actions": {
        "actions": [
            {
                "name": "Read"
            },
            {
                "name": "Write"
            },
            {
                "name": "Delete"
            }
        ]
    }
}

```

## Related Information

[CapabilityType \[page 221\]](#)

## 12.4 Sensors and Things

Describes the role and relationship of sensors and things

### Overview

Speaking of things in the Internet of Things only makes sense if you extend the focus to the sensors that are attached to the things you want to control or monitor in an IoT scenario. It is the sensors that are the very source of the data that can be collected in the form of timeseries data. Here is how the various objects involved come together during thing onboarding.

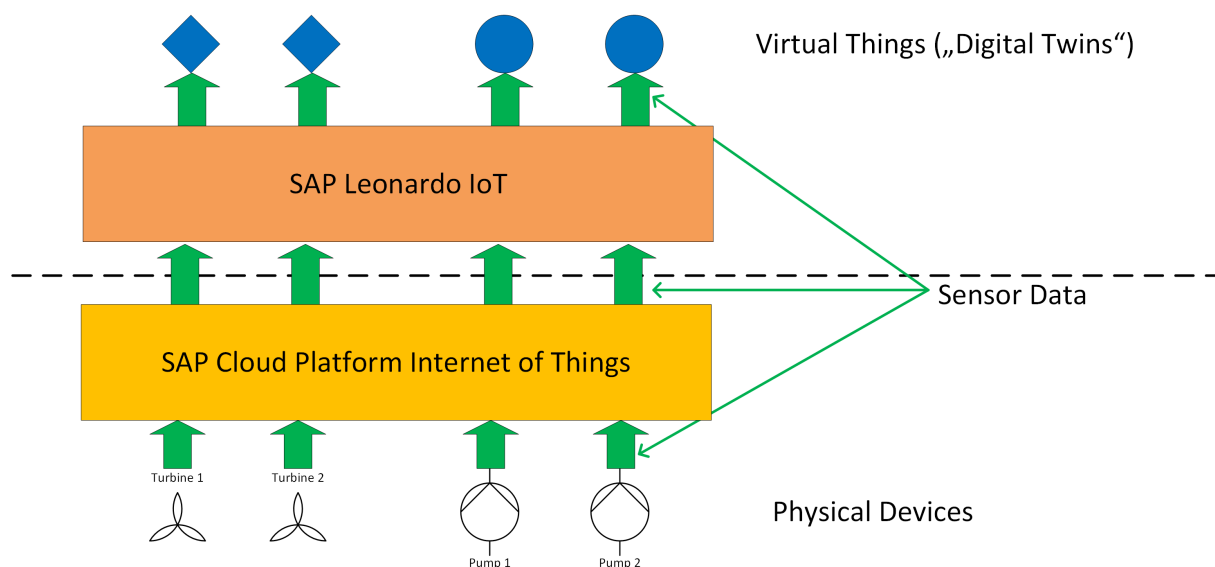


## Onboarding Steps

Onboarding of a thing consists of the following activities:

- Identifying the physical devices to be covered in an IoT scenario.
- For each type of device: Defining a thing type with a property set that reflects the different kinds of sensor data that the device can provide. During thing type definition, the system automatically scans the device model and tries to match the sensors found in the device model with the properties that belong to the thing type.
- Create a device model, that is, a software layer (typically provided by SAP Cloud Platform Internet of Things) that mirrors the sensor types and sensors that are part of a physical device.
- For each device: Define a thing based on the previously defined thing types.

Here is a diagram showing the various object levels and software layers involved:



As the diagram suggests, each of the physical devices sends one stream of sensor data through the different layers to the virtual things modeled with SAP Leonardo IoT.

### i Note

It may be surprising at first sight that the underlying architecture is designed such that each thing can only have one sensor assigned. However, that does **not** mean that for one thing, you could only collect sensor data of just one type, for example, temperature or pressure, but not both.

Actually, each sensor can be understood as an entity with a complex inner structure. Because of that, each single sensor can provide different types of data streams in parallel.

The services that you find grouped together in this section are dealing with retrieving the devices and sensors that have been brought into the reach of SAP Leonardo IoT and how the mapping between sensors and things can be accomplished. With that, you have programmatic access to an important part of an IoT scenario that is otherwise available only during creation of a thing in the Thing Modeler app.

In contrast to almost all other services offered by SAP Leonardo IoT, the services related to sensors and things are very limited in their functional scope and don't provide the complete set of CRUD methods.

## Mapping of Data Types Provided by SAP Cloud Platform Internet of Things for the Cloud Foundry Environment to SAP Leonardo IoT Data Types

SAP Leonardo IoT has no immediate access to physical devices. Rather, it relies on the SAP Cloud Platform Internet of Things layer that is in charge of providing a unified access to sensor-enabled hardware of all kinds. However, the type system of SAP Cloud Platform Internet of Things for the Cloud Foundry Environment is slightly different from the one that has been implemented for SAP Leonardo IoT. For this reason, the data types of both layers must be mapped. The mapping itself is taken care of by the system, so there is no need for manual activities here. The following table explains which types correspond to each other:

SAP Leonardo IoT	SAP Cloud Platform Internet of Things for the Cloud Foundry Environment	Comment
Numeric	integer	
Numeric(s,n)	double, float	May lead to a loss of precision if the received value does not fit it the specified scale and precision
NumericFlexible	double, float	May lead to a loss of precision if the received value does not fit it the specified scale and precision
String	string	
Boolean	Boolean	
Date	date, long	
DateTime	date, long	
Timestamp	date, long	
JSON	Not supported	
LargeString	string	
ByteArray	Not supported	
Not supported	binary	
ThingID	string	
BusinessPartnerID	string	
GeoJSON	string	

### Related Information

<https://help.sap.com/viewer/18e345f32da643fc90c029facb5c0248/latest/en-US/d1eacdde3e2e48499bfb11cbdaf8d7de.html>

## 12.4.1 Flexible Mapping (OData)

Flexible approach of mapping sensor properties to thing properties

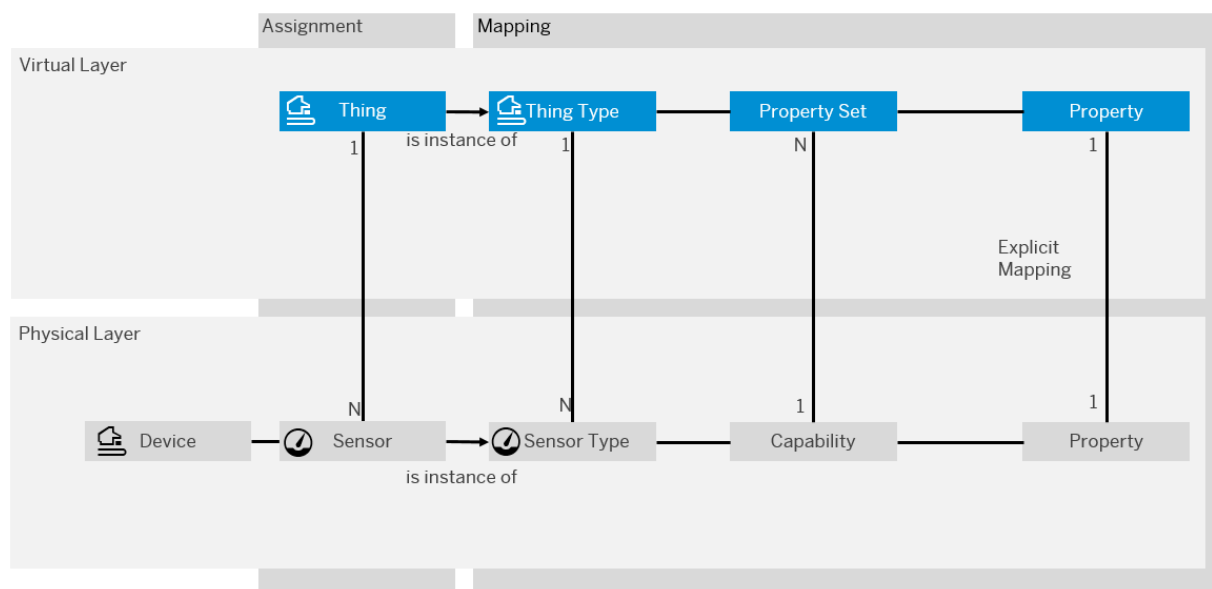
### Overview

When it comes to mapping the sensor properties of a physical device to the properties of a virtual thing, earlier versions of SAP Leonardo IoT have followed a name-based approach. This means that a sensor property could only be mapped to a thing property of the same data type and the same name.

SAP Leonardo IoT now offers a much more powerful approach for the same purpose named "flexible mapping". As the name suggests, flexible mapping gives you much more freedom and flexibility with regard to the way you can model the relationship between the physical and the virtual layer of an IoT scenario.

Flexible mapping gives you a lot of freedom when it comes to modeling decisions with regards to mapping sensor properties to thing properties. The flexible approach supports a much more refined data model compared to the name-based approach that was offered in earlier releases of SAP Leonardo IoT.

### Architecture



Flexible Mapping (OData) - Thing and Device Model

In the above image, you see that some of the relationships have been extended from 1:1 to 1:N (which increases the reuse potential of the affected entities) and that additional entities (assignments and sensor types) appear in the device model. You can also see the difference between mapping (type level) and assignment (instance level).

## Features

Here is a list of the most important and powerful features that let flexible mapping stand out compared to name-based mapping:

- Architectural distinction between entities on type level and on instance level
- Supports mapping of sensor properties and thing properties without having to ensure that equal names are used on both layers.
- Supports the mapping of properties belonging to multiple sensor types to the properties of one thing type (n:1).
- Supports the definition of multiple different mappings for the same thing type.
- Sensor properties that belong to different device capabilities can be mapped to the properties of multiple property sets.
- Supports human-readable master data for mappings so that users don't have to bother about highly technical IDs.

## Limitations

Although we have chosen to call the new approach of mapping device capabilities to thing properties "flexible mapping", that doesn't mean that everything is possible. Instead, the following architectural basic conditions apply:

- Sensor properties that belong to different device capabilities cannot be mapped to the properties of a single thing property set.
- One sensor type cannot be mapped to a particular thing type more than once.

### ❖ Example

Sensor type ST1 may comprise various individual sensors for different value dimensions (such as pressure, temperature, acceleration). You might want to install the same sensor type at various positions of a machine where you want to use different subsets of sensors at each position. However, this is currently **not** supported.

- A particular set of sensor types can be mapped only once to the same thing type.

### ❖ Example

Sensor types ST1, ST2 and ST3 have been mapped to thing type TT1. After that, you are free to define additional mappings for TT1 such as (TT1 - ST1), (TT1 - ST2), (TT1 - ST2, ST3). However, it is not allowed to define a second mapping (TT1 - ST1, ST2, ST3) because this particular combination already exists.

- The master data fields of a mapping (**<Name>**, **<Description>**) are **not** part of any kind of translation workflow. The texts that you maintain for these fields will appear the same, regardless of your logon language.

### i Note

Regardless of this limitation, you can still do something to support multiple languages: You may maintain multiple **<language>:<description text>** pairs in the **<Description>** field, for

example: "en": "Fan", "de": "Ventilator", "fr": "Ventilateur". It is then up to your applications accessing the mapping at runtime to pick the text in the language that matches best the current browser or user profile settings.

## Related Information

[Credentials \[page 753\]](#)

[Mapping \[page 761\]](#)

[Assignment \[page 785\]](#)

### 12.4.1.1 Credentials

Service for authenticating SAP Leonardo IoT accesses to objects residing on the SAP Cloud Platform Internet of Things for the Cloud Foundry Environment layer

When an application based on SAP Leonardo IoT tries to associate a thing with the sensors of a physical device, it depends on the resources offered via the SAP Cloud Platform Internet of Things for the Cloud Foundry Environment layer. Since this layer comes with a user and authorization management of its own, the application has to verify that it is authorized to access device and sensor data. This service offers the methods that are needed for that purpose.

#### i Note

The Credentials service is needed only in scenarios where the target value feature for thing attributes is used (for more information, see [Thing: Reference Property Data \[page 858\]](#)). In contrast to that, the service is **not** needed for scenarios where sensor data is only used for analytical purposes.

The Credentials service always operates within the scope of the tenant you are logged on to. There is no support for cross-tenant operations with this service.

With the latest release of SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (4.28.0), the authorization data in the Credentials service is updated to support IoT services tenant and instance specific details.

#### Base URI:

- Formal description: `http://<server address>[:<port number>]/v1/credentials`
- Example for a base URI in a cloud foundry environment: `https://tm-data-mapping.cfapps.eu10.hana.ondemand.com/v1/credentials`

**Permissions:** none

## Methods

HTTP Method	Action	URI	Scopes
<i>POST</i>	<a href="#">Create Credentials [page 754]</a>	/v1/credentials	n/a
<i>GET</i>	<a href="#">Read Credentials [page 757]</a>	/v1/credentials	n/a
PUT	<a href="#">Update Credentials [page 758]</a>	/v1/credentials	n/a
<i>DELETE</i>	<a href="#">Delete Credentials [page 760]</a>	/v1/credentials	n/a

## Payload

For POST, PUT, and GET, only media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "user"      : { "type": "string", "maxLength": 256, "required": "true" },
  "password"  : { "type": "string", "maxLength": 256, "required": "true" },
  "host"      : { "type": "string", "maxLength": 128, "required": "true" },
  "instanceId": { "type": "string", "maxLength": 55, "required": "false" },
  "iotTenantId": { "type": "string", "maxLength": 55, "required": "false" }
}
```

### 12.4.1.1.1 Create Credentials

Setting the credentials for accessing sensor data

With this service method, you define the authentication data that are used by an application based on SAP Leonardo IoT to access sensor data provided by a physical device via the SAP Cloud Platform Internet of Things for the Cloud Foundry Environment layer.

## Request

**URI:** /v1/credentials

**HTTP Method:** *POST*

**Permissions:** none.

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
user	Yes	string	User ID for logging on to the SAP Cloud Platform Internet of Things layer.	Request body
password	Yes	string	Password for logging on to the SAP Cloud Platform Internet of Things layer.	Request body
host	Yes	string	Landscape-specific ID string of the SAP Cloud Platform Internet of Things service instance to connect to.	Request body
instanceId	No	string	Instance ID for accessing SAP Cloud Platform Internet of Things service version 4.28.0 or above	Request body
iotTenantId	No	string	Tenant for accessing SAP Cloud Platform Internet of Things service version 4.28.0 or above, to support tenant-specific enhancements	Request body

## Request Example

/v1/credentials

The following request payload is used to create credentials if you are using SAP Cloud Platform Internet of Things for the Cloud Foundry Environment version older than 4.28.0.

```
{
  "user": "root",
  "password": "pass1",
  "host": "kafkatest-sandbox.canary.cp.iot.sap"
}
```

/v1/credentials

The following request payload is used to create credentials if you are using SAP Cloud Platform Internet of Things for the Cloud Foundry Environment version 4.28.0 and above.

```
{
  "user": "root",
  "password": "pass1",
  "host": "kafkatest-sandbox.canary.cp.iot.sap",
  "instanceId": "kafkatest-sandbox",
  "iotTenantId": "1450145528"
}
```

## Payload

For POST, PUT, and GET, only media type JSON is supported. The JSON for these methods has the following structure:

```
{
```

```

    "user"      : { "type": "string", "maxLength": 256, "required": "true" },
    "password"  : { "type": "string", "maxLength": 256, "required": "true" },
    "host"      : { "type": "string", "maxLength": 128, "required": "true" },
    "instanceId": { "type": "string", "maxLength": 55, "required": "false" },
    "iotTenantId": { "type": "string", "maxLength": 55, "required": "false" }
  }

```

## Response

### Response Headers

Header	Description
location	URL pointing to the newly created Credentials instance
ETag	Entity tag of the newly created Credentials instance

### Response Status and Error Codes

Code	Description
201	Credentials created successfully
400	Request could not be processed due to invalid or missing parameters
409	Credentials passed by the method exist already

### Response Example

The following is the response payload after creating the credentials, if you are using SAP Cloud Platform Internet of Things for the Cloud Foundry Environment version older than 4.28.0.

```

{
  "user": "root",
  "tenantId": "test-sandbox",
  "host": "kafkatest-sandbox.canary.cp.iot.sap"
}

```

The following is the response payload after creating the credentials, if you are using SAP Cloud Platform Internet of Things for the Cloud Foundry Environment version 4.28.0 and above.

```

{
  "user": "root",
  "tenantId": "test-sandbox",
  "host": "kafkatest-sandbox.canary.cp.iot.sap",
  "instanceId": "kafkatest-sandbox",
  "iotTenantId": "1450145528"
}

```

## Related Information

[Credentials \[page 753\]](#)



## 12.4.1.1.2 Read Credentials

Retrieving the credentials for accessing sensor data

With this service method, you retrieve the authentication data that are required by an application based on SAP Leonardo IoT to access sensor data provided by a physical device via the SAP Cloud Platform Internet of Things for the Cloud Foundry Environment layer. The actual value of the credentials data is defined by the SAP Cloud Platform Internet of Things for the Cloud Foundry Environment layer, and it is specific for each tenant.

### i Note

With the latest release of SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (4.28.0), the authorization data includes tenant ID and instance ID in the response payload, if you provided these values while creating the credentials.

## Request

**URI:** `/v1/credentials`

**HTTP Method:** [GET](#)

**Permissions:** none.

### Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

### Request Parameters

None.

### Request Example

`/credentials`

With this request, all the credentials for the current tenant are returned.

## Response

### Response Headers

Header	Description
Authorization	JWT (JSON Web Token)
ETag	Entity tag of the retrieved Credentials object

## Response Status and Error Codes

Code	Description
200	Credentials retrieved successfully
404	Not Found - Request successfully processed, no return value (e.g., no credentials available for the specified tenant)

## Response Example

The following is the response payload after creating the credentials, if you are using SAP Cloud Platform Internet of Things for the Cloud Foundry Environment older than 4.28.0.

```
{
  "user": "root",
  "tenantId": "test-sandbox",
  "host": "kafkatest-sandbox.canary.cp.iot.sap"
}
```

The following is the response payload after creating the credentials, if you are using SAP Cloud Platform Internet of Things for the Cloud Foundry Environment version 4.28.0 and above.

```
{
  "user": "root",
  "tenantId": "test-sandbox",
  "host": "kafkatest-sandbox.canary.cp.iot.sap",
  "instanceId": "kafkatest-sandbox",
  "iotTenantId": "1450145528"
}
```

## Related Information

[Credentials \[page 753\]](#)

### 12.4.1.1.3 Update Credentials

With this method, you change the credentials needed for accessing data provided by the SAP Cloud Platform Internet of Things for the Cloud Foundry Environment layer

You use this method to change the credentials needed to access the sensor data provided by the SAP Cloud Platform Internet of Things for the Cloud Foundry Environment layer. This is useful to adapt the service to a changed situation with regards to the credentials in effect. For example, a company-wide security policy may require that the credentials must be changed after a fixed period of time to protect the system against unauthorized access. Such changes in the underlying layer must then be reflected by calling this method.

## Request

URI: /v1/credentials

HTTP Method: *PUT*

Permissions: none.

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)
If-Match	Yes	Check if cached version matches server version of the credentials

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
user	Yes	string	User ID for logging on to the SAP Cloud Platform Internet of Things layer.	Request body
password	Yes	string	Password for logging on to the SAP Cloud Platform Internet of Things layer.	Request body
host	Yes	string	Landscape-specific ID string of the SAP Cloud Platform Internet of Things service instance to connect to.	Request body
instanceId	No	string	Instance ID for accessing SAP Cloud Platform Internet of Things service version 4.28.0 or above	Request body
iotTenantId	No	string	Tenant for accessing SAP Cloud Platform Internet of Things service version 4.28.0 or above, to support tenant-specific enhancements	Request body

## Request Example

/v1/credentials

```
{
  "user": "root",
  "password": "pass2",
  "host": "kafkatest-sandbox.canary.cp.iot.sap"
}
```

### Note

If you are using SAP Cloud Platform Internet of Things for the Cloud Foundry Environment version 4.28.0 and above, you must also include the parameters `instanceId` and `iotTenantId` in the request payload.

## Payload

For POST, PUT, and GET, only media type JSON is supported. The JSON for these methods has the following structure:

```
{
  "user"      : { "type": "string", "maxLength": 256, "required": "true" },
  "password"  : { "type": "string", "maxLength": 256, "required": "true" },
  "host"      : { "type": "string", "maxLength": 128, "required": "true" },
  "instanceId": { "type": "string", "maxLength": 55, "required": "false" },
  "iotTenantId": { "type": "string", "maxLength": 55, "required": "false" }
}
```

## Response

### Response Headers

Header	Description
ETag	Entity tag of the updated credentials

### Response Status and Error Codes

Code	Description
200	Credentials updated successfully

## Related Information

[Credentials \[page 753\]](#)

### 12.4.1.1.4 Delete Credentials

Deleting the credentials for accessing sensor data

With this service method, you delete the authentication data that are in effect for an application based on SAP Leonardo IoT to access sensor data provided by a physical device via the SAP Cloud Platform Internet of Things for the Cloud Foundry Environment layer. Once the credentials have been deleted with this method, none of the applications that exist in the specified tenant can access sensor data of the specified data provider. This is because the credentials data is valid for an entire tenant.

## Request

URI: `/v1/credentials`

HTTP Method: *DELETE*

Permissions: none.

### Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)
If-Match	Yes	Check if cached version matches server version of the credentials

### Request Parameters

None.

### Request Example

`/v1/credentials`

With this request, you delete the credentials that are in effect for granting the current tenant `sap-iotaedemo` access to sensor data provided by the default device data provider.

## Response

### Response Status and Error Codes

Code	Description
204	Credentials deleted successfully

## Related Information

[Delete Credentials \[page 760\]](#)

## 12.4.1.2 Mapping

Service for managing the mapping of a thing to a physical device on type level

With this service, you can set up a connection between the sensor types of a physical device and a virtual thing type that represents the device in an IoT scenario. To accomplish this, the devices must already be modeled with the respective methods provided by the SAP Cloud Platform Internet of Things layer. SAP Cloud Platform Internet of Things provides unified access to sensors and devices of various kinds, and it serves as a hardware abstraction layer for solutions like SAP Leonardo IoT.

## i Note

Defining mappings for sensor types and thing types is a prerequisite for the consecutive creation of assignments between sensors and things on instance level. For more information, see [Assignment \[page 785\]](#).

### Base URI:

- Formal description: `http://<server address>[:<port number>]/[path]/v1/Mappings`
- Example for a base URI in a cloud foundry environment: `https://tm-data-mapping.cfapps.eu10.hana.ondemand.com/v1/Mappings`

### Permissions:

`<thing>.r`

`<thingconf!t5>.r`

`<thing!t5>.r`

`<thing!t5>.conf.r`

## Methods

HTTP Method	Action	URI	Scopes
POST	<a href="#">Create a Mapping [page 763]</a>	<code>/v1/Mappings</code>	<code>&lt;thing&gt;.r</code>
GET	<a href="#">Read a Mapping [page 767]</a>	<code>/v1/Mappings/&lt;mappingID&gt;</code>	<code>&lt;thing&gt;.r</code>
GET	<a href="#">Read all Mapping Master Data [page 768]</a>	<code>/v1/Mappings/MasterData</code>	<code>&lt;thing&gt;.r</code>
GET	<a href="#">Read all Mapping IDs [page 770]</a>	<code>/v1/Mappings/mappingIds</code>	<code>&lt;thing&gt;.r</code>
PUT	<a href="#">Update a Mapping [page 772]</a>	<code>/v1/Mappings/&lt;mappingID&gt;</code>	<code>&lt;thing&gt;.r</code>
PUT	<a href="#">Update Mapping Master Data [page 774]</a>	<code>/v1/Mappings/&lt;mappingID&gt;/MasterData</code>	<code>&lt;thing&gt;.r</code>
DELETE	<a href="#">Delete a Mapping [page 776]</a>	<code>/v1/Mappings/&lt;mappingID&gt;</code>	<code>&lt;thing&gt;.r</code>
GET	<a href="#">Validate a Mapping [page 777]</a>	<code>/v1/Mappings/&lt;mappingID&gt;/validate</code>	<code>&lt;thing&gt;.r</code>
GET	<a href="#">Propose Mappings [page 779]</a>	<code>/v1/Mappings/proposal?thingTypeId='&lt;ThingTypeID&gt;'&amp;sensorTypeId='&lt;SensorTypeID&gt;' /v1/Mappings/proposal?thingId='&lt;ThingID&gt;'&amp;sensorId='&lt;SensorID&gt;'</code>	<code>&lt;thing&gt;.r</code>

## Related Information

## 12.4.1.2.1 Create a Mapping

With this method, you associate a thing type with a sensor type.

After a physical device has been made available by SAP Cloud Platform Internet of Things for higher-level software layers like SAP Leonardo IoT, you can establish a mapping between the sensor properties of the device and the properties of the virtual thing. This procedure is often referred to as "onboarding a thing". To accomplish this, you execute this service method and pass the sensor type ID, the thing type ID as well as the actual property mapping in the payload of the method.

Although technically not required, we recommend making use of the non-mandatory master data fields `<name>` and `<description>`. They are meant for providing a place where human-readable texts can be kept describing the purpose of a particular mapping. All other identifying attributes in this context are highly technical and non-intuitive, so your life will be easier when you take this opportunity.

### i Note

Successful mapping of device sensors to things depends on a number of prerequisites. For more information, see [Requirements for Successful Thing Onboarding](#).

## Mapping of Data Types Provided by SAP Cloud Platform Internet of Things for the Cloud Foundry Environment to SAP Leonardo IoT Data Types

SAP Leonardo IoT has no immediate access to physical devices. Rather, it relies on the SAP Cloud Platform Internet of Things layer that is in charge of providing a unified access to sensor-enabled hardware of all kinds. However, the type system of SAP Cloud Platform Internet of Things for the Cloud Foundry Environment is slightly different from the one that has been implemented for SAP Leonardo IoT. For this reason, the data types of both layers must be mapped. The mapping itself is taken care of by the system, so there is no need for manual activities here. The following table explains which types correspond to each other:

SAP Leonardo IoT	SAP Cloud Platform Internet of Things for the Cloud Foundry Environment	Comment
Numeric	integer	
Numeric(s,n)	double, float	May lead to a loss of precision if the received value does not fit it the specified scale and precision
NumericFlexible	double, float	May lead to a loss of precision if the received value does not fit it the specified scale and precision
String	string	
Boolean	Boolean	

SAP Leonardo IoT	SAP Cloud Platform Internet of Things for the Cloud Foundry Environment	Comment
Date	date, long	
DateTime	date, long	
Timestamp	date, long	
JSON	Not supported	
LargeString	string	
ByteArray	Not supported	
Not supported	binary	
ThingID	string	
BusinessPartnerID	string	
GeoJSON	string	

## Request

**URI:** /v1/Mappings

**HTTP Method:** *POST*

**Permissions:**

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
name	No	string(max. 127 characters)	Name of the mapping	Request body
description	No	string (max 255 characters)	Short description of the mapping	Request body
thingTypeId	Yes	string	Technical ID of the thing type to be mapped to a sensor type	Request body



Parameter	Required	Data Type	Description	Parameter Type
sensorTypeId	Yes	string	Technical ID of the sensor type to be mapped to a thing type	Request body
(measures   targetValues).capabilityId	Yes	string		Request body
(measures   targetValues).propertyMappings	Yes	array of objects	Set of mappings between physical device capabilities and virtual thing type properties; 1 or more mappings required; applies to either measured values or target values	Request body
(measures   targetValues).propertyMappings.CapabilityPropertyId	Yes	string	Device (sensor) capability to be mapped to a property; applies to either measured values or target values	Request body
(measures.propertyMappings.npstPropertyId   targetValues.propertyMappings.npstRefPropertyId)	Yes	string	Property to be mapped to a device (sensor) capability; applies to either measured values or target values	Request body

### Note

If you want to map sensor data to a property in a property set that has been defined for calculated values, you can use the `(measures.propertyMappings.<...>` parameters for that. These parameters can handle both measured values as well as calculated values.

### Request Example

```
{
  "name": "My First Mapping",
  "description": {
    "en": "My test description",
    "de": "Meine Testbeschreibung"
  },
  "thingTypeId": "com.iotae.demo.pidi:CoffeeMachine",
  "mappings": [
    {
      "sensorTypeId": "75128d4c-b52e-4a10-8b8f-46ad81917fc5",
      "measures": [
```

```

{
  "capabilityId": "576",
  "namedPropertySetId": "Fill_Level_Water",
  "propertyMappings": [
    {
      "capabilityPropertyId": "9dda2b90-00af-485a-bdc0-02888ef73dfc",
      "npstPropertyId": "Temperature"
    }
  ]
},
{
  "targetValues": [
    {
      "capabilityId": "581",
      "namedPropertySetId": "Brewing_Unit",
      "propertyMappings": [
        {
          "capabilityPropertyId": "f1f2f276-c9a3-4c0e-a3cb-d9823e6baa72",
          "npstRefPropertyId": "Pressure"
        }
      ]
    }
  ]
}
]
}

```

The example above demonstrates how you can create a mapping between the properties of a sensor type and the properties of a thing type. In this example, there are actually two different kinds of mappings being established simultaneously:

- One mapping for measured values (read out the current value measured by a sensor): Temperature
- One mapping for target values (set a desired value for the device): Pressure

Since all the values are passed on to the service via arrays, you could as well establish as many mappings as you like (or, as there are capabilities and properties available) in one single call of the service method.

## Response

### Response Status and Error Codes

Code	Description
201	New mapping created successfully
400	Bad request (for example, wrong or missing parameters)

## Related Information

[Mapping \[page 761\]](#)

## 12.4.1.2.2 Read a Mapping

Retrieves the mapping data for the mapping with the specified ID from the database

You call this method to retrieve the technical details of the mapping with the specified ID. You typically call this method after retrieving the list of mapping IDs that belong to a particular thing type with the help of the [Read all Mapping IDs \[page 770\]](#) service method.

### Request

**URI:** /v1/Mappings/<mappingID>

**HTTP Method:** *GET*

**Permissions:**

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

### Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
mappingID	Yes	string	Technical ID of the mapping	Path

### Request Example

```
GET /v1/Mappings/6fc8f297-431a-43f7-a42e-dbf584c76594
```

### Response

#### Response Status and Error Codes

Code	Description
200	Mapping retrieved successfully

Code	Description
400	Bad request. Request could not be processed due to invalid or missing parameters

## Response Example

```
{
  "name": "My First Mapping",
  "description": {
    "en": "My test description"
  },
  "thingTypeId": "com.iotae.demo.pidi:CoffeeMachine",
  "mappings": [
    {
      "sensorTypeId": "75128d4c-b52e-4a10-8b8f-46ad81917fc5",
      "measures": [
        {
          "capabilityId": "576",
          "namedPropertySetId": "Fill_Level_Water",
          "propertyMappings": [
            {
              "capabilityPropertyId": "9dda2b90-00af-485a-bdc0-02888ef73dfc",
              "npstPropertyId": "Temperature"
            }
          ]
        }
      ]
    }
  ],
  "targetValues": []
},
"etag": 1
}
```

In the above example, the mapping returned for the specified ID contains exactly one pair of sensor type capability/thing type property, namely the measured value for the temperature dimension.

## Related Information

[Mapping \[page 761\]](#)

[Read all Mapping IDs \[page 770\]](#)

### 12.4.1.2.3 Read all Mapping Master Data

Retrieves the master data of a mapping for the specified object ID

With this service method, you retrieve the master data (that is, the mandatory ID as well as the non-mandatory name and description) of the mapping objects that belong to the specified IDs. As input parameter, you may provide both sensor type IDs as well as thing type IDs.

## Request

**URI:** /v1/Mappings/MasterData

**HTTP Method:** [GET](#)

**Permissions:**

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
thingTypeId	No	string	ID of the thing type for which you want to retrieve mappings.	Query string
sensorTypeId	No	string	ID of the sensor type for which you want to retrieve mappings.	Query string
\$skip	No	integer	Number of the first n records to be excluded from the result set	Query string
\$top	No	integer	Number of records to include in the result set	Query string

### Note

If the \$top parameter is omitted, the method returns up to 1000 entries. You can increase the parameter to a maximum value of 10000. If this is not sufficient, you need to call the service method multiple times.

## Request Example

```
GET /v1/Mappings/MasterData?thingTypeId eq 'sap-dev.demoPackage:ThingType1'
```

## Response

### Response Status and Error Codes

Code	Description
200	Master data retrieved successfully
400	Bad request. Request could not be processed due to invalid or missing parameters

### Response Example

```
[
  {
    "id" : "6fc8f297-431a-43f7-a42e-dbf584c76594",
    "name" : "Default Mapping",
    "description" : {
      "en": "Default mapping for thing type"
    }
    "thingTypeId" : "sap-dev.demoPackage:ThingType1",
    "sensorTypeIds" : ["SensorType1", "SensorType2"],
    "etag": 1
  }
]
```

In the above example, the method returns exactly one mapping object where the properties of the thing type passed as input parameter are mapped to the capabilities of two different sensor types.

## Related Information

[Mapping \[page 761\]](#)

### 12.4.1.2.4 Read all Mapping IDs

Retrieves the IDs of all mappings defined for the specified thing type or sensor type

With this method, you can retrieve the complete list of mapping IDs that have been created for the thing type, or sensor type, respectively, passed as input parameter to the method.

Unlike most of the other "Read all..." methods in the SAP Leonardo IoT framework, this method is **not** technically limited to a maximum number of objects to be retrieved with one call. This is because the number of useful mappings has a "natural" limit derived from the number of supported properties or device capabilities. These numbers fall almost always into a range that is not critical in terms of server load.

If no mappings exist, the method returns an empty array in its payload.

#### i Note

You may call this method by passing the ID of either a sensor type or a thing type, but not both.

## Request

**URI:** /v1/Mappings/mappingIds

**HTTP Method:** *GET*

**Permissions:**

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
thingTypeId	No	string	ID of the thing type	Query
sensorTypeId	No	string	ID of the sensor type	Query

### Note

Although, from a technical perspective, both `thingTypeId` as well as `sensorTypeId` are flagged as **not** required, you still need to provide exactly one of them when you call this service method. That is, the two parameters are mutually exclusive.

If the `$top` parameter is omitted, the method returns up to 1000 entries. You can increase the parameter to a maximum value of 10000. If this is not sufficient, you need to call the service method multiple times.

## Request Example

```
GET /v1/Mappings/mappingIds?thingTypeId=com.iotae.demo.pidi:CoffeeMachine
GET /v1/Mappings/mappingIds?sensorTypeId=75128d4c-b52e-4a10-8b8f-46ad81917fc5
```

The above example illustrates the two different options of calling this service method by either passing a thing type ID or a sensor type ID. Under the assumption that the thing type and the sensor type mentioned are mapped to each other, both flavors of this method call yield the same result.

## Response

### Response Status and Error Codes

Code	Description
200	Mappings retrieved successfully, <b>or</b> No mapping found for the specified ID (empty array).

### Response Example

```
[  
  "75128d4c-b52e-4a10-8b8f-46ad81917fc5",  
  "f1f2f276-c9a3-4c0e-a3cb-d9823e6baa72"  
]
```

In the above example, the method has returned two different mapping IDs for the specified object type. You could now use the returned IDs as input for the [Read a Mapping \[page 767\]](#) method. Keep in mind that each of the returned mapping objects may contain up to two different sets of mappings (measured values or target values) between sensor type properties and thing type properties.

## Related Information

[Mapping \[page 761\]](#)

[Read a Mapping \[page 767\]](#)

## 12.4.1.2.5 Update a Mapping

With this method, you modify the technical details of a mapping relationship

You use this method to update the technical details of a mapping by directly modifying the fields in the method payload. In other words, you can access and change all of the data that has been stored for a mapping, such as sensor type, capabilities, properties, and so on. Of course, you need to take care that after the update, the properties referenced by the mapping still match (for example, no type conflicts) and make sense from a business perspective.

### i Note

When updating a mapping, you always need to provide the entire payload, not only the fields to be updated. Otherwise, fields that are **not** provided will be reset to their default values.

While this service method focuses on the technical details of a mapping, you can also modify the master data of a mapping. To accomplish this, use the [Update Mapping Master Data \[page 774\]](#) method.



## Request

**URI:** /v1/Mappings/<mappingID>

**HTTP Method:** *PUT*

**Permissions:**

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (Java Web Token)
If-Match	Yes	Check if cached version matches server version of the Mapping object

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
mappingID	Yes	string	Technical identifier of the mapping to be updated	Path
Payload	Yes	JSON	Complete payload of the mapping containing the updated field values	Request body

## Request Example

```
PUT /v1/Mappings/6fc8f297-431a-43f7-a42e-dbf584c76594
{
  "name": "My First Mapping",
  "description": {
    "en": "My test description"
  },
  "thingTypeId": "com.iotae.demo.pidi:CoffeeMachine",
  "mappings": [
    {
      "sensorTypeId": "75128d4c-b52e-4a10-8b8f-46ad81917fc5",
      "measures": [
        {
          "capabilityId": "576",
          "namedPropertySetId": "Fill_Level_Water",
          "propertyMappings": [
            {
              "capabilityPropertyId": "9dda2b90-00af-485a-bdc0-02888ef73dfc",
              "npstPropertyId": "Temperature"
            }
          ]
        }
      ]
    }
  ],
  "targetValues": [
    {
      "capabilityId": "581",
      "namedPropertySetId": "Brewing_Unit",
```

```

    "propertyMappings": [
      {
        "capabilityPropertyId": "f1f2f276-c9a3-4c0e-a3cb-d9823e6baa72",
        "npstRefPropertyId": "Pressure"
      }
    ]
  }
]
}

```

Modifies the mapping with the specified ID and replaces its field values according to the payload provided in the request body.

## Response

### Response Status and Error Codes

Code	Description
200	Mapping updated successfully

## Related Information

[Mapping \[page 761\]](#)

[Update Mapping Master Data \[page 774\]](#)

### 12.4.1.2.6 Update Mapping Master Data

With this method, you modify the master data of a mapping relationship

You use this method to update the master data of a mapping by directly modifying the `<name>` and `<description>` fields in the method payload. This is especially helpful when you find that the highly technical field values stored for the mapped entities do not really help you much in understanding what a particular mapping is about.

#### i Note

When updating a mapping, you always need to provide the entire payload, not only the fields to be updated. Otherwise, fields that are **not** provided will be reset to their default values.

While this service method focuses on the master data of a mapping, you can also modify the technical details of a mapping. To accomplish this, use the [Update a Mapping \[page 772\]](#) method.

## Request

**URI:** /v1/Mappings/<mappingID>/MasterData

**HTTP Method:** *PUT*

**Permissions:**

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (Java Web Token)
If-Match	Yes	Check if cached version matches server version of the Mapping object

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
mappingID	Yes	string	Technical identifier of the mapping to be updated	Path
Payload	Yes	JSON	Complete payload of the mapping containing the updated field values	Request body

## Request Example

```
PUT /v1/Mappings/6fc8f297-431a-43f7-a42e-dbf584c76594/MasterData
```

Modifies the mapping with the specified ID and replaces its field values according to the payload provided in the request body.

## Response

### Response Status and Error Codes

Code	Description
200	Mapping updated successfully

## Response Example

```
{
  "id" : "6fc8f297-431a-43f7-a42e-dbf584c76594",
  "name" : "Default Mapping",
  "description" : {
```

```

    "en": "Default mapping for thing type"
  }
  "thingTypeId" : "sap-dev.demoPackage:ThingType1",
  "sensorTypeIds" : ["SensorType1", "SensorType2"],
  "etag": 1
}

```

## Related Information

[Mapping \[page 761\]](#)

[Update a Mapping \[page 772\]](#)

### 12.4.1.2.7 Delete a Mapping

With this method, you delete the mapping between a thing type and its sensor types

You use this method to delete a sensor/thing mapping on type level. After deleting a mapping, all sensor types that were part of the mapping are no longer available to the thing type.

#### Note

You can only delete a mapping on type level after all corresponding sensor thing assignments on instance level have been deleted.

## Request

**URI:** /v1/Mappings/<mappingID>

**HTTP Method:** *DELETE*

**Permissions:**

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

Header	Required	Values
If-Match	Yes	Check if cached version matches server version of the Mapping object

#### i Note

Instead of supplying a particular ETag value of the assignment to be deleted, you can also provide a wildcard ("\*"). Using the wildcard ensures that the assignment is deleted in any case, even if it has been modified by a concurrent process.

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
mappingID	Yes	string	Technical identifier of the mapping to be deleted	Path

## Request Example

```
DELETE /v1/Mappings/6fc8f297-431a-43f7-a42e-dbf584c76594
```

## Response

### Response Status and Error Codes

Code	Description
204	Mapping deleted successfully.
400	Bad request (for example, no mapping ID provided)
404	Not found (for example, the mapping has already been deleted in the meantime; may be avoided by providing ETag = * as request parameter)

## Related Information

[Mapping \[page 761\]](#)

[Delete an Assignment \[page 794\]](#)

## 12.4.1.2.8 Validate a Mapping

Helps you determine the root source of data ingestion errors

In a Big Data business scenario that you have set up with the tools and services of SAP Leonardo IoT it is not unusual that from time to time you will encounter error log entries. Such entries indicate that the sensors

attached to your monitored things have sent data, but something has gone wrong during the data ingestion process, that is, the transfer of incoming data streams into the various persistence layers in the backend of the SAP Leonardo IoT infrastructure.

One potential root cause for such errors are changes to the involved data structures used in a given thing and device model, such as renamed fields, field type modifications, etc. The `validate` service method analyzes the error log for the specified mapping and makes you aware of all errors that were caused by data incompatibilities between sender and receiver within the SAP Leonardo IoT infrastructure.

## Request

**URI:** `/v1/Mappings/<mappingID>/validate`

**HTTP Method:** `GET`

**Permissions:**

`<thing>.r`

`<thingconf!t5>.r`

`<thing!t5>.r`

`<thing!t5>.conf.r`

### Request Headers

Header	Required	Values
Authorization	Yes	JWT (Java Web Token)

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
mappingID	Yes	string	Technical identifier of the mapping to be validated	Path

### Request Example

```
GET /v1/Mappings/6fc8f297-431a-43f7-a42e-dbf584c76594/validate
```

## Response

### Response Status and Error Codes

Code	Description
200	Validation executed successfully

### i Note

Even if errors are detected during the validation, the service method still returns HTTP status code 200 for "successful execution". The "success" here refers to executing the validation, but **not** necessarily to the formal correctness of the mapping that has been validated. Formal errors found in the mapping are reflected by error messages that the method returns as its payload in the response body (see the following example).

### Response Example

```
[
  "Could not find thing type with ID:
  de.vtest.test.tmdmpostman.odata:ThingTypeValidate"
]
```

### Related Information

[Mapping \[page 761\]](#)

[Error Messages \[page 796\]](#)

## 12.4.1.2.9 Propose Mappings

Lets the system analyze sensor capabilities and thing properties and propose a mapping

In earlier releases of SAP Leonardo IoT, the only way to set up a mapping between the properties of physical devices and virtual things was to find pairs of properties in both objects involved with the same name and data type. While the flexible mapping introduced with version 1.62 is no longer limited to only accept property pairs with matching names, you can still have the system present a proposal of matching properties based on these criteria. Creating such a proposal is the purpose of this service method.

The method can be parameterized in two different ways:

- type-based (that is, thing type and sensor type)
- instance-based (that is, thing and sensor)

After execution, the method returns the calculated mapping proposal in the response payload. It is then up to you to either modify or accept the proposed mapping and store it in the database by handing the payload over to the [Create a Mapping \[page 763\]](#) method.

### Request

URI:

a) /v1/Mappings/proposal?thingTypeId='<ThingTypeID>'&sensorTypeId='<SensorTypeID>'

b) /v1/Mappings/proposal?thingId='<ThingID>'&sensorId='<SensorID>'

HTTP Method: [GET](#)

#### Permissions:

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

#### Request Headers

Header	Required	Values
Authorization	Yes	JWT (Java Web Token)

#### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
thingTypeId	No	string	Technical identifier of the thing type involved in the mapping.	Query string
sensorTypeId	No	string	Technical identifier of the sensor type involved in the mapping.	Query string
thingId	No	string	Technical identifier of the thing involved in the mapping.	Query string
sensorId	No	string	Technical identifier of the sensor involved in the mapping.	Query string

#### Note

Although, from a technical perspective, all the parameters listed above are flagged as **not** required, you still need to provide exactly one pair of them when you call this service method (either type-based or instance-based). That is, the two pairs of parameters are mutually exclusive. Also, it is **not** allowed to combine a parameter referring to an instance with another referring to a type.

#### Request Example

```
a) GET /v1/Mappings/proposal?
thingTypeId=com.iotae.demo.pidi:CoffeeMachine&sensorTypeId=75128d4c-
b52e-4a10-8b8f-46ad81917fc5
b) GET /v1/Mappings/proposal?thingId=6fc8f297-431a-43f7-a42e-
dbf584c76594&sensorId=8976
```

The above example demonstrates both flavors of calling the proposal method, either type-based (a) or instance-based (b).



## Response

### Response Status and Error Codes

Code	Description
200	Mapping proposal created successfully

### Response Example

```
{
  "thingTypeId": "com.iotae.demo.pidi:CoffeeMachine",
  "mappings": [
    {
      "sensorTypeId": "75128d4c-b52e-4a10-8b8f-46ad81917fc5",
      "measures": [
        {
          "capabilityId": "576",
          "namedPropertySetId": "Fill_Level_Water",
          "propertyMappings": [
            {
              "capabilityPropertyId": "9dda2b90-00af-485a-bdc0-02888ef73dfc",
              "npstPropertyId": "Temperature"
            }
          ]
        }
      ]
    },
    {
      "targetValues": [
        {
          "capabilityId": "581",
          "namedPropertySetId": "Brewing_Unit",
          "propertyMappings": [
            {
              "capabilityPropertyId": "f1f2f276-c9a3-4c0e-a3cb-d9823e6baa72",
              "npstRefPropertyId": "Pressure"
            }
          ]
        }
      ]
    }
  ]
}
```

In the above example, the method has proposed the mapping of two sensor type capabilities to two corresponding properties of a thing type. Moreover, the properties fall into two different technical categories (`<measures>` and `<targetValues>`).

## Related Information

[Mapping \[page 761\]](#)

[Create a Mapping \[page 763\]](#)

## 12.4.1.2.10 Read all Assignments for a Mapping

Retrieves a list of all sensor/thing assignments based on the specified mapping.

With this method, you can retrieve a list of all sensor/thing assignments on instance level that have been created based on the specified sensor type/thing type mapping on type level. You may want to use the resulting list for analytical purposes or as a preliminary step for a cleanup action, when you want to get rid of a particular mapping but have to delete the referring assignments first.

### i Note

Unlike the collective GET request method of the [Assignment \[page 785\]](#) service (see the *Related Information* section below), this method is always focused on the assignments related to a particular mapping.

## Request

**URI:** /v1/Mappings/<mappingID>/assignments

**HTTP Method:** *GET*

**Permissions:**

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
mappingID	Yes	string	Unique ID of the mapping for which you want to retrieve the existing assignments	Path

## Request Example

```
GET /v1/Mappings/6fc8f297-431a-43f7-a42e-dbf584c76594/assignments
```

The request above returns a list of all sensor/thing assignments on instance-level that have been created by reference to the mapping with the specified ID.

## Response

### Response Status and Error Codes

Code	Description
200	Assignment retrieved successfully

### Response Example

```
[
  {
    "id": "f38fac75-7e65-4953-86a9-92330adf4e3a",
    "thingId": "A68BCF0D32914C19AACD1225AF887330",
    "sensorIds": ["220", "221"],
    "mappingId": "6fc8f297-431a-43f7-a42e-dbf584c76594",
    "ETag": 1
  }
]
```

From the response payload given in the above example, you can see that for the specified mapping, there is only one assignment (represented by its `id`). Along with the assignment, the method also provides the thing and sensors involved.

## Related Information

[Mapping \[page 761\]](#)

[Delete all Assignments for a Mapping \[page 783\]](#)

[Read all Assignments \[page 790\]](#)

### 12.4.1.2.11 Delete all Assignments for a Mapping

Deletes all sensor/thing assignments based on the specified mapping.

With this method, you can delete all sensor/thing assignments on instance level that have been made by reference to the specified mapping. This is useful when you want to carry out a cleanup activity for a mapping that is no longer needed. Calling this method should be preceded by calling the [Read all Assignments for a Mapping \[page 782\]](#) method, which lets you get an overview of all affected assignments.

#### i Note

This method is different compared to almost all other Delete methods that are available in the API services of SAP Leonardo IoT because it aims at deleting an entire set of assignments at once rather than focusing on one single assignment.

## Request

**URI:** /v1/Mappings/<mappingID>/assignments

**HTTP Method:** *DELETE*

**Permissions:**

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
mappingID	Yes	string	Unique ID of the mapping for which you want to retrieve the existing assignments	Path

## Request Example

```
/v1/Mappings/6fc8f297-431a-43f7-a42e-dbf584c76594/assignments
```

In this example, all assignments related to the mapping with the specified ID are deleted. After successful execution, you can delete the specified mapping itself.

## Response

### Response Status and Error Codes

Code	Description
204	Assignments deleted successfully.
400	Bad request (for example, no mapping ID provided)
404	Not found (for example, the assignments have already been deleted in the meantime; may be avoided by providing ETag = * as request parameter)

## Related Information

[Mapping \[page 761\]](#)

[Read all Assignments for a Mapping \[page 782\]](#)

### 12.4.1.3 Assignment

Service for managing the mapping of a thing to a physical device on instance level

With this service, you can set up a connection between the sensors of a physical device and a virtual thing that represents the device in an IoT scenario. To accomplish this, the devices must already be modeled with the respective methods provided by the SAP Cloud Platform Internet of Things layer. SAP Cloud Platform Internet of Things provides unified access to sensors and devices of various kinds, and it serves as a hardware abstraction layer for solutions like SAP Leonardo IoT.

#### Note

Defining assignments for sensors and things on instance level requires that a mapping of the involved thing types and sensor types has already been created.

#### Base URI:

- Formal description: `https://<server address>[:<port number>][<path>]/v1/Assignments`
- Example for a base URI in a cloud foundry environment: `https://tm-data-mapping.cfapps.eu10.hana.ondemand.com/v1/Assignments`

#### Permissions:

`<thing>.r`

`<thingconf!t5>.r`

`<thing!t5>.r`

`<thing!t5>.conf.r`

## Methods

HTTP Method	Action	URI	Scopes
POST	<a href="#">Create an Assignment [page 786]</a>	<code>/v1/Assignments</code>	<code>&lt;thing&gt;.r</code>
GET	<a href="#">Read an Assignment [page 789]</a>	<code>/v1/Assignments/&lt;assignmentID&gt;</code>	<code>&lt;thing&gt;.r</code>
GET	<a href="#">Read all Assignments [page 790]</a>	<code>/v1/Assignments</code>	<code>&lt;thing&gt;.r</code>
PUT	<a href="#">Update an Assignment [page 792]</a>	<code>/v1/Assignments/&lt;assignmentID&gt;</code>	<code>&lt;thing&gt;.r</code>

HTTP Method	Action	URI	Scopes
DELETE	<a href="#">Delete an Assignment [page 794]</a>	/v1/Assignments/<assignmentID>	<thing>.r

## Related Information

[Mapping \[page 761\]](#)

### 12.4.1.3.1 Create an Assignment

With this method, you associate a thing with a sensor

After a physical device has been made available by SAP Cloud Platform Internet of Things for higher-level software layers like SAP Leonardo IoT, you can establish an assignment between the properties of the virtual thing and the sensor properties of the device. This procedure is often referred to as "onboarding a thing". To accomplish this, you execute this service method and pass the sensor ID as well as the thing ID in the payload of the method. In addition, you need to provide the ID of a mapping between the corresponding sensor types and thing type.

You can use this method not only to create one assignment per call. Instead, it is possible to pass an array of many assignments in one call. This is useful, for example, when you have to provide sensor/thing assignments for an entire shop floor at once (as opposed to a continual onboarding of things over a longer period of time). This kind of mass operations can be done with the same service method by simply providing all the pairs of properties and capabilities at once in the payload of the request.

#### i Note

Calling this service method with a payload for multiple assignments is limited to a maximum of 250 assignments per call.

#### Prerequisite

To create an assignment on instance level, there must be a mapping referencing the superordinate types of the sensor and thing involved.

## Request

**URI:** /v1/Assignments

**HTTP Method:** *POST*

**Permissions:**

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
thingId	Yes	string	Technical ID of the thing to be mapped to one or more sensors	Request body
sensorIds	Yes	string	Array of technical IDs of the sensors to be mapped to a thing	Request body
mappingId	Yes	string	Technical ID of the sensor type/thing type mapping on type level	Request body

## Request Example

```
POST /v1/Assignments
{
  "thingId": "A68BCF0D32914C19AACD1225AF887330",
  "sensorIds": ["220", "221"],
  "mappingId": "adfed57a-a537-4078-813b-ee8381e9fe26"
}
```

In the above example, the thing with the specified ID is assigned to two sensors (220, 221). The specified `<mappingID>` points to a mapping object where the respective sensor type is mentioned from which the two sensors have been derived.

```
POST /v1/Assignments
[
  {
    "thingId": "A68BCF0D32914C19AACD1225AF887330",
    "sensorIds": ["220", "221"],
    "mappingId": "adfed57a-a537-4078-813b-ee8381e9fe26"
  },
  {
    "thingId": "B68BCF0D32914C19AACD1225AF887330",
    "sensorIds": ["222", "223"],
    "mappingId": "adfed57a-a537-4078-813b-ee8381e9fe26"
  }
]
```

In the above example, four sensors (220, 221, 222, 223) are assigned to two different things. The fact that the specified `<mappingID>` is the same for both mappings indicates that both things are instances of the same thing type.

## Response

### Response Headers

Header	Description
location	URL path to the newly created assignment
ETag	Entity tag used to identify a particular instance of an assignment

### Response Status and Error Codes

Code	Description
201	Assignment created successfully
400	Bad Request - Request could not be processed due to invalid or missing parameters

### Response Example

```
POST /v1/Assignments
{
  "thingId": "A68BCF0D32914C19AACD1225AF887330",
  "sensorIds": ["220", "221"],
  "mappingId": "adfed57a-a537-4078-813b-ee8381e9fe26"
}
```

In the above example, you see the result payload for a POST request for only one single assignment to be created. For this flavor of calling the service method, the ETag value is **not** part of the payload. Rather, it is returned in the `<ETag>` header field.

```
POST /v1/Assignments
[
  {
    "id": "f38fac75-7e65-4953-86a9-92330adf4e3a",
    "thingId": "A68BCF0D32914C19AACD1225AF887330",
    "sensorIds": ["220", "221"],
    "mappingId": "adfed57a-a537-4078-813b-ee8381e9fe26",
    "ETag": 1
  },
  {
    "id": "f38fac75-7e65-4953-86a9-92330adf4e3a",
    "thingId": "B68BCF0D32914C19AACD1225AF887330",
    "sensorIds": ["222", "223"],
    "mappingId": "adfed57a-a537-4078-813b-ee8381e9fe26",
    "ETag": 2
  }
]
```

In the above response payload, you see the result for the simultaneous creation of two assignments with one call. Note that for this execution mode, the correct `<ETag>` value of the created assignments can only be read from the payload. This is for the simple reason that the `<ETag>` header field can of course only hold one value at a time, not many.



## Related Information

[Assignment \[page 785\]](#)

[Create a Mapping \[page 763\]](#)

### 12.4.1.3.2 Read an Assignment

Retrieves the assignment data for the assignment with the specified ID from the database

You call this method to retrieve the technical details of the assignment with the specified ID. You typically call this method after having retrieved the list of assignmentIDs that belong to a particular thing or sensor with the help of the [Read all Assignments \[page 790\]](#) service method.

## Request

**URI:** `/v1/Assignments/<assignmentID>`

**HTTP Method:** *GET*

**Permissions:**

`<thing>.r`

`<thingconf!t5>.r`

`<thing!t5>.r`

`<thing!t5>.conf.r`

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
assignment ID	Yes	string	[Description, possible values, default values]	Path

## Request Example

```
GET /v1/Assignments/f38fac75-7e65-4953-86a9-92330adf4e3a
```

## Response

### Response Headers

Header	Description
ETag	Entity tag used to identify a particular instance of an assignment

### Response Status and Error Codes

Code	Description
200	Assignment retrieved successfully
404	Not found - Requested resource was not found

### Response Example

```
{
  "id": "f38fac75-7e65-4953-86a9-92330adf4e3a",
  "thingId": "A68BCF0D32914C19AACD1225AF887330",
  "sensorIds": ["220", "221"],
  "mappingId": "adfed57a-a537-4078-813b-ee8381e9fe26"
}
```

## Related Information

[Assignment \[page 785\]](#)

[Read all Assignments \[page 790\]](#)

### 12.4.1.3.3 Read all Assignments

Retrieves the all assignments defined for the specified thing or sensor

With this method, you can retrieve the complete list of assignments that have been created for the thing, or sensor, respectively, passed as input parameter to the method. Note that the two parameters should be considered mutually exclusive. You can also omit both parameters. In that case, the system returns a complete list of all existing assignments on instance level.

## Request

URI: `/v1/Assignments`

HTTP Method: `GET`

Permissions:

<thing>.r  
<thingconf!t5>.r  
<thing!t5>.r  
<thing!t5>.conf.r

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
thingId	No	string	Unique identifier of the thing for which assignments shall be retrieved	Query string
sensorId	No	string	Unique identifier of the sensor for which assignments shall be retrieved	Query string
\$skip	No	integer	Number of the first n records to be excluded from the result set	Query string
\$top	No	integer	Number of records to include in the result set	Query string

### Note

Although, from a technical perspective, both `thingId` as well as `sensorId` are flagged as **not** required, you still should provide only one of them when you call this service method. That is, the two parameters should be considered mutually exclusive. If, however, you decide to provide both parameters simultaneously, you should be aware that you can always accomplish the same result with a more simple request. In addition, if the combined values for both parameters don't match with any record in the database, the method returns an empty result.

If the `$top` parameter is omitted, the method returns up to 1000 entries. You can increase the parameter to a maximum value of 10000. If this is not sufficient, you need to call the service method multiple times.

## Request Example

```
GET /v1/Assignments
GET /v1/Assignments?thingId=A68BCF0D32914C19AACD1225AF887330
GET /v1/Assignments?sensorId=220
```

The above example shows the three different flavors of reading assignments:

- Unfiltered request
- Read assignments for a specified thing
- Read assignments for a specified sensor

## Response

### Response Status and Error Codes

Code	Description
200	Assignments retrieved successfully, <b>or</b> No mapping found for the specified ID (empty array).

### Response Example

```
GET /v1/Assignments?thingId=A68BCF0D32914C19AACD1225AF887330
[
  {
    "id": "f38fac75-7e65-4953-86a9-92330adf4e3a",
    "thingId": "A68BCF0D32914C19AACD1225AF887330",
    "sensorIds": ["220", "221"],
    "mappingId": "adfed57a-a537-4078-813b-ee8381e9fe26",
    "ETag": 1
  }
]
```

## Related Information

[Assignment \[page 785\]](#)

## 12.4.1.3.4 Update an Assignment

With this method, you modify the technical details of a sensor/thing assignment

You use this method to update the technical details of an assignment by directly modifying the fields in the method payload. In other words, you can access and change all of the data that has been stored for an assignment, such as the list of sensors assigned, the thing as well as the referenced mapping on type level. Of course, you need to take care that after the update, the entities referenced by the assignment still match (for example, no type conflicts) and make sense from a business perspective.

### i Note

This service method supports updating of only one single assignment per call. That is, you cannot provide an array of multiple assignments in the request payload.

## Request

URI: /v1/Assignments/<assignmentID>

HTTP Method: *PUT*

### Permissions:

<thing>.r

<thingconf!t5>.r

<thing!t5>.r

<thing!t5>.conf.r

### Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)
If-Match	Yes	Check if cached version matches server version of the Assignment object

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
assignment ID	Yes	string	Unique identifier of the assignment to be updated	Path
Payload	Yes	JSON	Request payload containing all updated values	Request body

### Request Example

```
PUT /v1/Assignments/f38fac75-7e65-4953-86a9-92330adf4e3a
{
  "thingId": "A68BCF0D32914C19AACD1225AF887330",
  "sensorIds": ["210", "211", "212"],
  "mappingId": "adfed57a-a537-4078-813b-ee8381e9fe26"
}
```

In the above example, three sensors are assigned to the specified thing.

## Response

### Response Headers

Header	Description
ETag	Entity tag of the updated Assignment object

### Response Status and Error Codes

Code	Description
200	Assignment updated successfully

### Response Example

```
{
```

```
{
  "id": "f38fac75-7e65-4953-86a9-92330adf4e3a",
  "thingId": "A68BCF0D32914C19AACD1225AF887330",
  "sensorIds": ["210", "211", "212"],
  "mappingId": "adfed57a-a537-4078-813b-ee8381e9fe26"
}
```

## Related Information

[Assignment \[page 785\]](#)

### 12.4.1.3.5 Delete an Assignment

With this method, you delete one or more sensor/thing assignments

You use this method to delete existing assignments. For this, the method supports different calling conventions, which lets you decide to either delete a particular assignment specified by its ID or to delete all existing assignments for a particular thing.

## Request

URI:

- `/v1/Assignments/<assignmentID>`
- `/v1/Assignments?thingId=<thingID>`

HTTP Method: *DELETE*

Permissions:

`<thing>.r`

`<thingconf!t5>.r`

`<thing!t5>.r`

`<thing!t5>.conf.r`

## Request Headers

Header	Required	Values
Authorization	Yes	JWT (JSON Web Token)

Header	Required	Values
If-Match	Yes	Check if cached version matches server version of the Assignment object

### Note

Instead of supplying a particular ETag value of the assignment to be deleted, you can also provide a wildcard ("\*"). Using the wildcard ensures that the assignment is deleted in any case, even if it has been modified by a concurrent process.

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
assignment Id	Yes	string	Unique identifier of the assignment	Path
thingId	Yes	string	Unique identifier of the thing whose assignments shall be deleted	Query string

### Note

The two parameters cannot be combined in one call, that is, they are mutually exclusive. In addition, the syntax for the two ways of calling the method differs due to the different parameter type (path versus query string).

## Request Example

```
DELETE /v1/Assignments/f38fac75-7e65-4953-86a9-92330adf4e3a
DELETE /v1/Assignments?thingId=A68BCF0D32914C19AACD1225AF887330
```

The two DELETE requests in the example illustrate the two different flavors of this method:

- Deleting a particular assignment specified by its ID.
- Deleting all assignments that exist for the specified thing.

## Response

### Response Status and Error Codes

Code	Description
204	Assignment deleted successfully.
400	Bad request (for example, no assignment ID provided)
404	Not found (for example, the assignment has already been deleted in the meantime; may be avoided by providing ETag = * as request parameter)

## Related Information

[Assignment \[page 785\]](#)

### 12.4.2 Error Messages

A service that provides access to error messages

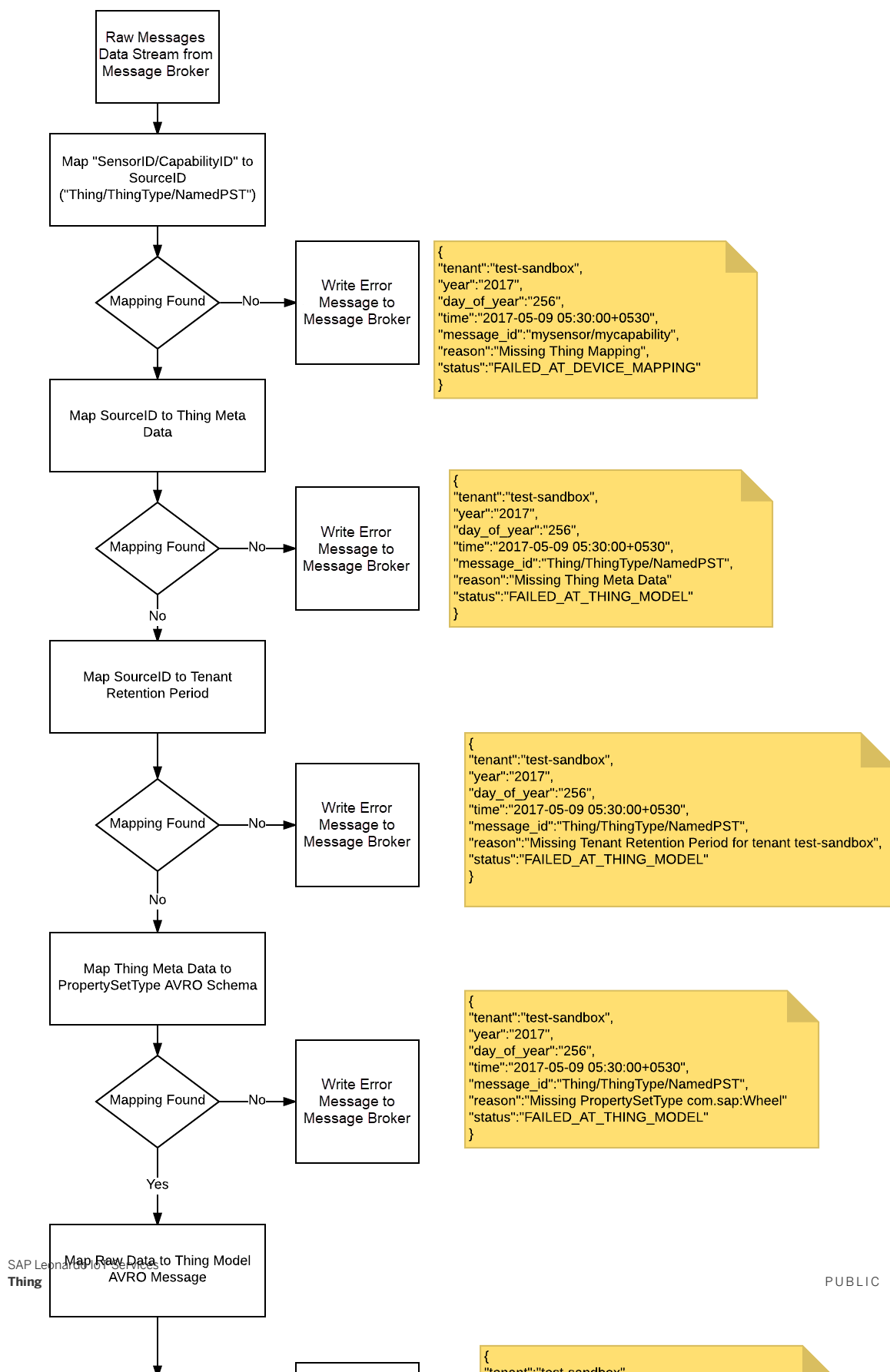
#### Overview

Like any other piece of software, SAP Leonardo IoT contains a set of several hundreds of error messages that are written to the error log whenever a service call cannot be executed successfully. A subset of these messages is dealing with problems that may occur during processing of incoming sensor data (typically, problems related to sensor/thing mapping or to the involved data types). Here, the system tries to collect as much information as possible to describe the root cause of the error as accurate as possible. This accuracy helps a system administrator to better understand the error situation and take appropriate action to further avoid that error.

There might be a situation when the time series data from the device is not persisted in the time series store for your tenant. Instead, an error message is written in the error log with status `FAILED_AT_TENANT_ONBOARDING`. This status indicates that the tenant was not onboarded successfully. Please raise a support ticket under the component `IOT-BSV-OPS-ONB`.



The following graphic illustrates an example of how the different system layers and components of SAP Leonardo IoT proceed to narrow down the set of potential error root causes (note how the `<reason>` part of the message changes from step to step as soon as more information is available):



## Base URI:

- Formal description: `http://<server address>[:<port number>][<path>]/v1/ErrorMessage`s
- Example for a base URI in a cloud foundry environment: `https://errorreader-dataingestion-sap.cfapps.eu10.hana.ondemand.com/v1/ErrorMessage`s

## Methods

This service consists of only one method for a collective GET request call into the set of error messages logged in the database during data ingestion. Since the number of logged errors may be high, the method may only be called for a specified time frame of no more than 24 hours.

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read all Error Messages [page 799]</a>	/v1/ErrorMessage	none

## Payload

Only media type JSON is supported. It has the following structure:

```
{
  "tenant"      : { "type" : "string" },
  "year"        : { "type" : "int"   },
  "day_of_year" : { "type" : "int"   },
  "time"        : { "type" : "timestamp" },
  "message_id"  : { "type" : "text"  },
  "status"      : { "type" : "text"  },
  "message"     : { "type" : "text"  },
  "reason"      : { "type" : "text"  }
}
```

## Data Protection and Privacy

In order to comply with applicable law regarding data protection and privacy, the system treats error messages according to the following rules:

- Error data that cannot be assigned to a property set clearly, is treated as non-personal data.
- If error data can be assigned to a property set, **and** if the property set is annotated as sensitive personal data (`com.sap.appiot.security:spi`), the actual ingested measured / time series data is masked. That is, you cannot re-ingest the data by reading it from the error log. Or, in other words, you can still see **that** data has been written, but the values cannot be restored.

### i Note

For more information on annotations for property sets, see [Annotation \[page 501\]](#) (OData).

## Related Information

[Sensors and Things \[page 748\]](#)

### 12.4.2.1 Read all Error Messages

Retrieves a list of error messages from the database

With this service method, you can retrieve a list of error messages from the system's error log database. In other words, you can find out about the error messages that have been logged during current operation because a service call has failed for some reason (especially due to problems related to sensor/thing mapping, or to the data types involved). This can be helpful in various scenarios:

- Tracking of a particular error that has occurred in the system.
- Providing error statistics.
- Analyzing the system's overall stability.

#### i Note

Retrieving high numbers of objects from the database with an unlimited collective GET request can lead to high system load and a significant increase of response times. Therefore, this method is limited such that with one call, only messages can be retrieved whose timestamps have the same calendar day in common. In addition, the specified timeframe must fall into the last seven days dating back from the moment the method is called. Error log entries older than that are automatically removed from the log and can no longer be retrieved.

This service method is **not** designed for retrieving a list of all the error messages that exist in the system. Also, there is no other method available serving the purpose of presenting a catalog of all existing error messages.

## Request

**URI:** `/v1/ErrorMessage`s

**HTTP Method:** [GET](#)

**Permissions:** none

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
<code>from</code>	No	Datetime	Period start for the messages to be retrieved	Path
<code>to</code>	No	Datetime	Period end for the messages to be retrieved	Path

Parameter	Required	Data Type	Description	Parameter Type
Authorization	Yes	JWT	Authorization for accessing sensor data	Header

### Note

According to the above table, the `from` and `to` parameters are qualified as **not** mandatory. However, this qualification has been given from a user perspective in the sense that it is allowed to omit these parameters in a method call.

However, from a **technical** perspective, providing values for these two parameters is still required. To overcome the apparent contradiction of a parameter qualified as mandatory and not mandatory at the same time, a default mechanism has been implemented, which works as follows:

- If none of the `from` and `to` parameters is passed to the system during a method call, the method yields the error messages that have been logged during the last hour before the method was called.
- If the `from` and `to` parameters are passed to the system, the method yields the error messages according to the specified timestamps.

## Request Example

```
/v1/ErrorMessage
```

Retrieves all the error messages related to data ingestion that have been logged during the last hour before the method call.

```
/v1/ErrorMessage?from=2017-10-25%2000:00:00Z&to=2017-10-25%2023:59:59Z
```

Retrieves all the error messages related to data ingestion that have been logged on October 25th, 2017. Note that in this example, the timespan defined by the `from` and `to` parameters exhausts the maximum duration allowed for this service method. Also, this code example triggers an error message when executed after November 1st, 2017 because that date would lie beyond the error log retention time of seven days.

## Response

### Response Status and Error Codes

Code	Description
200	Messages retrieved successfully
400	Bad Request. Request could not be processed due to invalid or missing parameters (for example, start and end date with different calendar days)

## Related Information

[Error Messages \[page 796\]](#)

## 12.5 Grouping of Things

### Overview

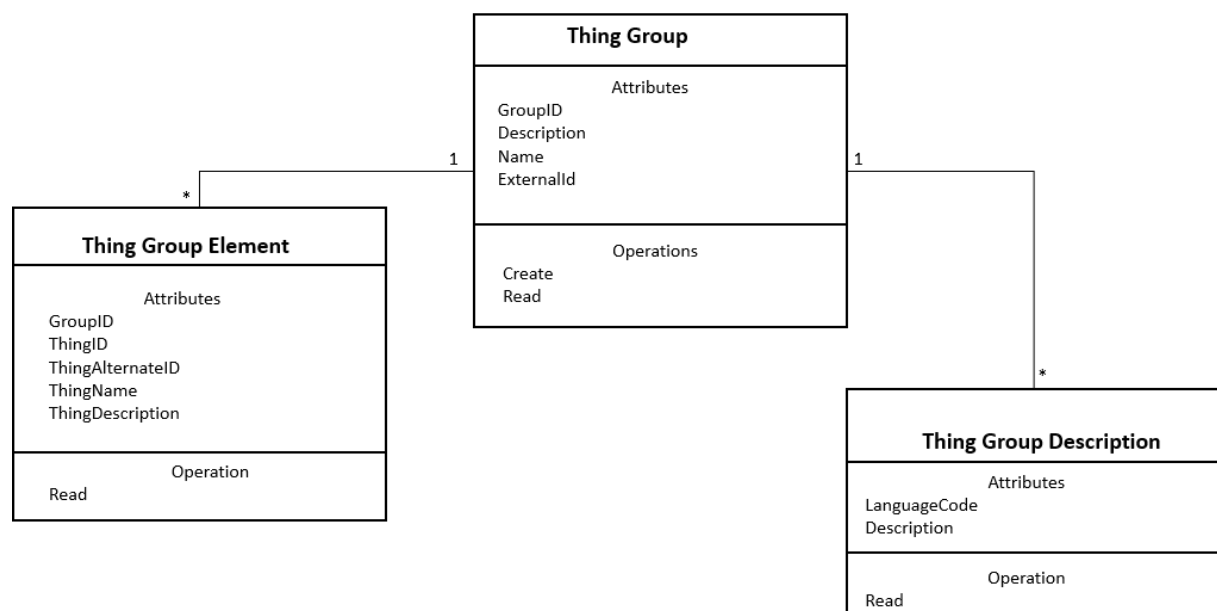
A group of things is similar to a single-level hierarchy. The things are grouped based on logical reason as per the business needs. You can create multiple groups and assign things to the groups. A thing can belong to multiple groups. For example, you can create a thing group with all things from a specific location. A user can maintain and monitor all things belonging to this group based on the business needs.

**OData Version: 2.0**

**Root URI:** `http://<server address>[:<port number>]/ThingGroup`

**Example for a Root URI in a cloud foundry environment:** `https://apiot-thing-group.cfapps.eu10.hana.ondemand.com/ThingGroup`

### Entity Data Model



**Service Metadata URI:** `http://<server address>[:<port number>]/ThingGroup/v1/$metadata`

### Resources

Resource	Description	Path
<a href="#">Thing Group [page 802]</a>	Represents group of things	<code>/v1/ThingGroups</code>

Resource	Description	Path
<a href="#">Thing Group Element [page 819]</a>	Thing group elements	/v1/ThingGroupElements  /v1/ThingGroups ('GroupId') ? \$expand=ThingGroupElements
<a href="#">Thing Group Description [page 822]</a>	Thing group description	/v1/ThingGroups ('GroupId') ? \$expand=ThingGroupDescriptions

## 12.5.1 Thing Group

Represents group of things

**Resource Path:** `http://<server address>[:<port number>]/ThingGroup/v1/ThingGroups`

### Operations

#### CRUD Operations

HTTP Method	Operation	URI	Scopes
<i>POST</i>	<a href="#">Create Thing Group [page 807]</a>	/ThingGroup/v1/ThingGroup	<tg>.c
<i>GET</i>	<a href="#">Read a Thing Group [page 809]</a>	/ThingGroup/v1/ThingGroups ('<thing group ID>')	<tg>.r
<i>GET</i>	<a href="#">Read all Thing Groups [page 813]</a>	/ThingGroup/v1/ThingGroups	<tg>.r
<i>DELETE</i>	<a href="#">Delete Thing Group [page 816]</a>	/ThingGroup/v1/ThingGroups ('<thing group ID>')	<tg>.d

### Thing Group Properties

Parameter	Data Type	Maximum Length	Description
GroupID	String	32	Unique identifier of the thing group generated by the system.

Parameter	Data Type	Maximum Length	Description
Name	String	255	<p>Name of the thing group</p> <div> <p><b>i Note</b></p> <p>Only the following characters are supported:</p> <ul style="list-style-type: none"> <li>Alphabets <b>a</b> through <b>z</b> and <b>A</b> through <b>Z</b></li> <li>Numeric digits <b>0</b> through <b>9</b></li> <li>Punctuation mark underscore (<b>_</b>)</li> </ul> </div>
Description	String	255	<p>Description of the thing group</p> <p>The descriptive text provided in this field is ignored and is replaced with the descriptive text provided in the <code>Description</code> field of the <b>Thing Group Description</b> entity.</p>
ExternalID	String	255	<p>Thing group identifier (which is not unique) from an external system. You can create multiple thing groups with the same external ID.</p> <div> <p><b>i Note</b></p> <p>Only the following characters are supported:</p> <ul style="list-style-type: none"> <li>Alphabets <b>a</b> through <b>z</b> and <b>A</b> through <b>Z</b></li> <li>Numeric digits <b>0</b> through <b>9</b></li> <li>Punctuation marks hyphen (<b>-</b>), underscore (<b>_</b>), colon (<b>:</b>), and full-stop (<b>.</b>)</li> </ul> </div>

## Related Information

## 12.5.1.1 Read Metadata of Thing Group

With this service, you can retrieve the metadata of thing group.

### Request

**URI:** /ThingGroup/v1/\$metadata

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <tg>.r

### Request Example

/ThingGroup/v1/\$metadata

### Response

#### Response Status and Error Codes

Code	Description
200	Metadata of thing group retrieved successfully

### Payload

**Format:** [XML](#)

```
<?xml version="1.0" ?>
<edmx:Edmx Version="1.0" xmlns:edmx="http://schemas.microsoft.com/ado/2007/06/edmx">
  <edmx:DataServices m:DataServiceVersion="1.0" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:sap="http://www.sap.com/Protocols/SAPData">
    <Schema Namespace="com.sap.appiot" xmlns="http://schemas.microsoft.com/ado/2008/09/edm">
      <EntityType Name="Description">
        <Documentation>
          <Summary>Details of Thing Group Description Entity Type</Summary>
        </Documentation>
        <Key>
          <PropertyRef Name="LanguageCode"></PropertyRef>
        </Key>
      </EntityType>
    </Schema>
  </DataServices>
</edmx:Edmx>
```



```

        </Key>
        <Property Name="LanguageCode" Type="Edm.String" Nullable="false"
MaxLength="2">
            <Documentation>
                <Summary>ISO code of the language of the thing hierarchy
description</Summary>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" MaxLength="255">
            <Documentation>
                <Summary>Description of the Thing Group</Summary>
            </Documentation>
        </Property>
    </EntityType>
    <EntityType Name="ThingGroup">
        <Documentation>
            <Summary>Details of Thing Group Entity Type</Summary>
        </Documentation>
        <Key>
            <PropertyRef Name="GroupID"></PropertyRef>
        </Key>
        <Property Name="GroupID" Type="Edm.String" Nullable="false"
MaxLength="32">
            <Documentation>
                <Summary>Unique identifier of the Thing Group</Summary>
                <LongDescription>It is automatically generated by the
system when creating Thing Group</LongDescription>
            </Documentation>
        </Property>
        <Property Name="Name" Type="Edm.String" Nullable="true"
MaxLength="255">
            <Documentation>
                <Summary>Name of the Thing Group</Summary>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="255" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Description of the Thing Group</Summary>
                <LongDescription>When creating a Thing Group, the text
provided in this field is ignored and replaced with the text provided in the
Description field of the Description entity</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ExternalID" Type="Edm.String" Nullable="true"
MaxLength="255">
            <Documentation>
                <Summary>Thing Group identifier (which is not unique)
from an external system </Summary>
            </Documentation>
        </Property>
        <NavigationProperty Name="ThingGroupElements"
Relationship="com.sap.apptot.ThingGroupToElementAssociation"
FromRole="ThingGroup" ToRole="ThingGroupElement"></NavigationProperty>
        <NavigationProperty Name="ThingGroupDescriptions"
Relationship="com.sap.apptot.ThingGroupToDescriptionAssociation"
FromRole="ThingGroup" ToRole="Description"></NavigationProperty>
    </EntityType>
    <EntityType Name="ThingGroupElement">
        <Documentation>
            <Summary>Details of Thing Group Element Entity Type</Summary>
        </Documentation>
        <Key>
            <PropertyRef Name="GroupID"></PropertyRef>
            <PropertyRef Name="ThingID"></PropertyRef>
        </Key>
        <Property Name="GroupID" Type="Edm.String" Nullable="false"
MaxLength="32">

```

```

        <Documentation>
            <Summary>Unique identifier of the Thing Group</Summary>
            <LongDescription>It is automatically generated by the
system when creating Thing Group</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingID" Type="Edm.String" Nullable="false"
MaxLength="32">
        <Documentation>
            <Summary>Unique identifier of the thing</Summary>
        </Documentation>
    </Property>
    <Property Name="ThingAlternateID" Type="Edm.String"
Nullable="true" MaxLength="32">
        <Documentation>
            <Summary>Alternate Id of the Thing</Summary>
        </Documentation>
    </Property>
    <Property Name="ThingName" Type="Edm.String" MaxLength="255"
sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Name of the thing specified when creating a
thing</Summary>
            <LongDescription>It cannot be edited during create
(POST) or update (PATCH) of a thing hierarchy.</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingDescription" Type="Edm.String"
MaxLength="60" sap:creatable="false" sap:updatable="false">
        <Documentation>
            <Summary>Description of the thing specified when
creating a thing</Summary>
            <LongDescription>It cannot be edited during create
(POST) or update (PATCH) of a thing hierarchy.</LongDescription>
        </Documentation>
    </Property>
    <Property Name="Operation" Type="Edm.String" MaxLength="6">
        <Documentation>
            <Summary>Operation to be performed on the thing
hierarchy element</Summary>
            <LongDescription>Indicates the add or delete operation
to be performed on the thing hierarchy element during PATCH request.</
LongDescription>
        </Documentation>
    </Property>
</EntityType>
<Association Name="ThingGroupToDescriptionAssociation">
    <End Type="com.sap.apptot.ThingGroup" Multiplicity="1"
Role="ThingGroup"></End>
    <End Type="com.sap.apptot.Description" Multiplicity="*"
Role="Description"></End>
</Association>
<Association Name="ThingGroupToElementAssociation">
    <End Type="com.sap.apptot.ThingGroup" Multiplicity="1"
Role="ThingGroup"></End>
    <End Type="com.sap.apptot.ThingGroupElement" Multiplicity="*"
Role="ThingGroupElement"></End>
</Association>
<EntityContainer Name="ThingGroupContainer"
m:IsDefaultEntityContainer="true">
    <EntitySet Name="Descriptions"
EntityType="com.sap.apptot.Description">
        <Documentation>
            <Summary>Details of Thing Group Descriptions Entity Set</
Summary>
        </Documentation>
    </EntitySet>

```

```

        <EntitySet Name="ThingGroups"
EntityType="com.sap.apiot.ThingGroup">
        <Documentation>
            <Summary>Details of Thing Group Entity Set</Summary>
        </Documentation>
    </EntitySet>
    <EntitySet Name="ThingGroupElements"
EntityType="com.sap.apiot.ThingGroupElement">
        <Documentation>
            <Summary>Details of Thing Group Elements Entity Set</
Summary>
        </Documentation>
    </EntitySet>
    <AssociationSet Name="ThingGroupToDescriptionAssociation"
Association="com.sap.apiot.ThingGroupToDescriptionAssociation">
        <End EntitySet="ThingGroups" Role="ThingGroup"></End>
        <End EntitySet="Descriptions" Role="Description"></End>
    </AssociationSet>
    <AssociationSet Name="ThingGroupToElementAssociation"
Association="com.sap.apiot.ThingGroupToElementAssociation">
        <End EntitySet="ThingGroups" Role="ThingGroup"></End>
        <End EntitySet="ThingGroupElements"
Role="ThingGroupElement"></End>
    </AssociationSet>
</EntityContainer>
</Schema>
</edmx:DataServices>
</edmx:Edmx>

```

## 12.5.1.2 Create Thing Group

With this service, you can create thing group for a specific tenant. You can maintain multiple thing groups for a specific tenant. A thing can belong to multiple groups. However, a thing cannot appear multiple times in the same group. The service allows you to create an empty thing group.

### i Note

Using a single POST method, you can create a thing group with less than 1000 things. You can only add those things for which you have authorized access.

## Request

**URI:** /ThingGroup/v1/ThingGroup

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** <tg>.c

## Request Example

/ThingGroup/v1/ThingGroup

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* structure for this method has the following structure.

```
{
  "Name": "ABCCarGroup",
  "ExternalID": "ABCCarGroup",
  "ThingGroupElements": [{
    "ThingID": "503B8711B5D0445688E7FF7ACFC2177A"
  },
  {
    "ThingID": "908758D995094A6F9CC72FC9C18BC366"
  }
],
  "ThingGroupDescriptions": [{
    "LanguageCode": "en",
    "Description": "Group of ABC Car Things"
  }, {
    "LanguageCode": "de",
    "Description": "Gruppe von ABC Car Things"
  }]
}
```

## Response

### Response Status and Error Codes

Code	Description
201	Thing group created successfully

### Response Payload Example

```
{
  "d": {
    "__metadata": {
      "id": "https://apiiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')",
      "uri": "https://apiiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')",
      "type": "com.sap.apptot.ThingGroup"
    },
    "GroupID": "3E930040F5E3F8021600B10200CB288F"
  }
}
```

## Related Information

[Thing Group \[page 802\]](#)

## 12.5.1.3 Read a Thing Group

Retrieves details of a thing group

With this service, you can retrieve the list of things that are part of the specified thing group. If you retrieve an empty group along with the `$expand` query parameter, the service only retrieves the group details.

### Request

**URI:** `/ThingGroup/v1/ThingGroups('<thing group ID>')`

**Operation Type:** CRUD

**HTTP Method:** [\*GET\*](#)

**Permissions:** `<tg>.r`

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
GroupId	Yes	String	Unique identifier of the thing group	Path

### Request Query Parameters

Parameter	Required	Data Type	Description
\$expand	No	String	Includes the related resources inline while retrieving the requested resources.  Supported associations of thing group that can be retrieved using <code>\$expand</code> are thing group elements and thing group description.
\$filter	No	String	Filter condition to be applied; all fields in the result set are valid.
\$format	No	String	Format to be applied on the result set.
\$select	No	String	Select condition for the data fields to be included in the result set; all fields in the result set are valid.

### Request Example

`/ThingGroup/v1/ThingGroups('3E930040F5E3F8021600B10200CB288F')`

## Response

### Response Status and Error Codes

Code	Description
200	Thing group details retrieved successfully

## Payload

Format: *JSON*

Media type supported are *JSON* and *XML*. The *JSON* for this method has the following structure.

### Example 1

```
/ThingGroup/v1/ThingGroups('3E930040F5E3F8021600B10200CB288F')
```

Retrieves the list of things in the group 3E930040F5E3F8021600B10200CB288F

```
{
  "d": {
    "__metadata": {
      "id": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')",
      "uri": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')",
      "type": "com.sap.appiot.ThingGroup"
    },
    "GroupID": "3E930040F5E3F8021600B10200CB288F",
    "Name": "ABCCarGroup",
    "Description": null,
    "ExternalID": "ABCCarGroup",
    "ThingGroupElements": {
      "__deferred": {
        "uri": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')/ThingGroupElements"
      }
    },
    "ThingGroupDescriptions": {
      "__deferred": {
        "uri": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')/ThingGroupDescriptions"
      }
    }
  }
}
```

### Example 2

```
/ThingGroup/v1/ThingGroups('3E930040F5E3F8021600B10200CB288F')?
```

```
$expand=ThingGroupDescriptions,ThingGroupElements
```

Retrieves the list of things in the group ABCGroup along with the thing group description and thing group elements

```
{
```

```

    "d": {
      "__metadata": {
        "id": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')",
        "uri": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')",
        "type": "com.sap.apiot.ThingGroup"
      },
      "GroupID": "3E930040F5E3F8021600B10200CB288F",
      "Name": "ABCCarGroup",
      "Description": "Group of ABC Car Things",
      "ExternalID": "ABCCarGroup",
      "ThingGroupElements": {
        "results": [
          {
            "GroupID": "3E930040F5E3F8021600B10200CB288F",
            "ThingID": "503B8711B5D0445688E7FF7ACFC2177A",
            "ThingAlternateID": null,
            "ThingName": "ABCXSeries",
            "ThingDescription": "ABC X Series",
            "Operation": null
          },
          {
            "GroupID": "3E930040F5E3F8021600B10200CB288F",
            "ThingID": "908758D995094A6F9CC72FC9C18BC366",
            "ThingAlternateID": null,
            "ThingName": "ABCZSeries",
            "ThingDescription": "ABC Z Series",
            "Operation": null
          }
        ]
      },
      "ThingGroupDescriptions": {
        "results": [
          {
            "LanguageCode": "en",
            "Description": "Group of ABC Car Things"
          },
          {
            "LanguageCode": "de",
            "Description": "Gruppe von ABC Car Things"
          }
        ]
      }
    }
  }
}

```

### Example 3

```

/ThingGroup/v1/ThingGroups('3E930040F5E3F8021600B10200CB288F')?
$expand=ThingGroupElements

```

Retrieves the list of things in the group 3E930040F5E3F8021600B10200CB288F along with the thing group elements

```

{
  "d": {
    "__metadata": {
      "id": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')",
      "uri": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')",
      "type": "com.sap.apiot.ThingGroup"
    },
    "GroupID": "3E930040F5E3F8021600B10200CB288F",
    "Name": "ABCCarGroup",

```

```

    "Description": "Group of ABC Car Things",
    "ExternalID": "ABCCarGroup",
    "ThingGroupElements": {
      "results": [
        {
          "GroupID": "3E930040F5E3F8021600B10200CB288F",
          "ThingID": "503B8711B5D0445688E7FF7ACFC2177A",
          "ThingAlternateID": null,
          "ThingName": "ABCXSeries",
          "ThingDescription": "ABC X Series",
          "Operation": null
        },
        {
          "GroupID": "3E930040F5E3F8021600B10200CB288F",
          "ThingID": "908758D995094A6F9CC72FC9C18BC366",
          "ThingAlternateID": null,
          "ThingName": "AB CZSeries",
          "ThingDescription": "ABC Z Series",
          "Operation": null
        }
      ]
    },
    "ThingGroupDescriptions": {
      "__deferred": {
        "uri": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')/ThingGroupDescriptions"
      }
    }
  }
}

```

#### Example 4

```

/ThingGroup/v1/ThingGroups('56930040F5E3F8021600B10200CB288F')?
$expand=ThingGroupElements

```

The group 56930040F5E3F8021600B10200CB288F is an empty group and result is as follows:

```

{
  "d": {
    "__metadata": {
      "id": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('56930040F5E3F8021600B10200CB288F')",
      "uri": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('56930040F5E3F8021600B10200CB288F')",
      "type": "com.sap.appiot.ThingGroup"
    },
    "GroupID": "56930040F5E3F8021600B10200CB288F",
    "Name": "EmptyGroup",
    "Description": null,
    "ExternalID": "EmptyGroup",
    "ThingGroupElements": {
      "__deferred": {
        "uri": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('56930040F5E3F8021600B10200CB288F')/ThingGroupElements"
      }
    },
    "ThingGroupDescriptions": {
      "__deferred": {
        "uri": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('56930040F5E3F8021600B10200CB288F')/ThingGroupDescriptions"
      }
    }
  }
}

```



```
}
```

## Related Information

[Thing Group \[page 802\]](#)

### 12.5.1.4 Read all Thing Groups

With this service, you can retrieve all thing groups to which the user has access.

In addition, you can retrieve a subset of all thing groups, according to the filter criteria provided. For a collective request for a set of thing groups, you have these options:

- You can restrict the set of thing groups to be retrieved by filter criteria based on the fields available in the response.
- You can define that the set of matching thing groups shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ThingGroup/v1/ThingGroups`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<tg>.r`

### Request Query Parameters

Parameter	Required	Data Type	Description
<code>\$filter</code>	No	String	Filter condition to be applied; all fields in the result set are valid.

Parameter	Required	Data Type	Description
\$inlinecount	No	String	<p>Total number of groups available; permissible values are <a href="#">allpages</a> and <a href="#">none</a>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"> <li>• If you use this with the <code>\$top</code> and <code>\$skip</code> parameters, the result contains the total count of groups available.</li> <li>• If you use this with the <code>\$filter</code> parameter, the result contains the total count of groups based on the defined filter condition.</li> </ul>
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; all fields in the result set are valid. To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$skip	No	Integer	Number of records to exclude from the result set.
\$top	No	Integer	Number of records to include in the result set.
\$select	No	String	Select condition for the data fields to be included in the result set; all fields in the result set are valid.
\$skiptoken	No	Integer	Identifies a starting point in the collection of entities identified by the URI containing the <code>\$skiptoken</code> parameter.
\$format	No	Integer	Format to be applied on the result set. For example, <a href="#">json</a> or <a href="#">xml</a> .

## Request Example

/ThingGroup/v1/ThingGroups

## Response

### Response Status and Error Codes

Code	Description
200	Details of all thing groups retrieved successfully

## Payload

Format: *JSON*

Media type supported are *JSON* and *XML*. The *JSON* for this method has the following structure.

### Example 1

/ThingGroup/v1/ThingGroups

Retrieves details of all thing groups

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')",
          "uri": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')",
          "type": "com.sap.appiot.ThingGroup"
        },
        "GroupID": "3E930040F5E3F8021600B10200CB288F",
        "Name": "ABCCarGroup",
        "Description": "Group of ABC Car Things",
        "ExternalID": "ABCCarGroup",
        "ThingGroupElements": {
          "__deferred": {
            "uri": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')/ThingGroupElements"
          }
        },
        "ThingGroupDescriptions": {
          "__deferred": {
            "uri": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('3E930040F5E3F8021600B10200CB288F')/ThingGroupDescriptions"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('45930040F5E3F8021600B10200CB288F')",
          "uri": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroup('45930040F5E3F8021600B10200CB288F')",
          "type": "com.sap.appiot.ThingGroup"
        }
      }
    ]
  }
}
```

```

    },
    "GroupID": "45930040F5E3F8021600B10200CB288F",
    "Name": "ABC2SCarGroup",
    "Description": "Group of ABC2S Car Things",
    "ExternalID": "ABC2SCarGroup",
    "ThingGroupElements": {
      "__deferred": {
        "uri": "https://appiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroup('45930040F5E3F8021600B10200CB288F')/ThingGroupElements"
      }
    },
    "ThingGroupDescriptions": {
      "__deferred": {
        "uri": "https://appiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroup('45930040F5E3F8021600B10200CB288F')/ThingGroupDescriptions"
      }
    }
  }, {
    "__metadata": {
      "id": "https://appiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroup('56930040F5E3F8021600B10200CB288F')",
      "uri": "https://appiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroup('56930040F5E3F8021600B10200CB288F')",
      "type": "com.sap.appiot.ThingGroup"
    },
    "GroupID": "56930040F5E3F8021600B10200CB288F",
    "Name": "EmptyGroup",
    "Description": "Group of Empty Car Things",
    "ExternalID": "EmptyGroup",
    "ThingGroupElements": {
      "__deferred": {
        "uri": "https://appiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroup('56930040F5E3F8021600B10200CB288F')/ThingGroupElements"
      }
    },
    "ThingGroupDescriptions": {
      "__deferred": {
        "uri": "https://appiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroup('56930040F5E3F8021600B10200CB288F')/ThingGroupDescriptions"
      }
    }
  }
]
}
}

```

## Related Information

[Thing Group \[page 802\]](#)

### 12.5.1.5 Delete Thing Group

With this service, you can delete a thing group with a specific thing group ID.

The ETag value is mandatory to delete a thing group. You must retrieve the ETag value of the thing group using the GET (/ThingGroup/v1/ThingGroups('<thing group ID>')) service. The ETag value is assigned to the request header parameter If-Match in the DELETE request. If you do not include the ETag value in the request header, the server sends HTTP response code 428. If the ETag value is not correct, the server sends HTTP response code 412 .

## Request

**URI:** /ThingGroup/v1/ThingGroups('<thing group ID>')

**Operation Type:** CRUD

**HTTP Method:** [DELETE](#)

**Permissions:** <tg>.d

### Request Header Parameters

Parameter	Required	Values
Accept-Language	No	Language of the thing type description. The default language is en.
If-Match	Yes	Latest ETag value of a specific thing group that is updated using the PATCH request.

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
GroupID	Yes	String	Unique identifier of the thing group	Path

### Request Example

/ThingGroup/v1/ThingGroups('50930040F5E3F8021600B10200CB288F')

## Response

### Response Status and Error Codes

Code	Description
204	Thing group deleted successfully

## Related Information

[Thing Group \[page 802\]](#)

## 12.5.1.6 Update Thing Group

With this service, you can add new things to the specified thing group or delete the existing things from the specified thing group.

The `ETag` value is mandatory to update a thing group. You must retrieve the `ETag` value of the thing group using the `GET (/ThingGroup/v1/ThingGroups ('<thing group ID>'))` service. The `ETag` value is assigned to the request header parameter `If-Match` in the `PATCH` request. If you do not include the `ETag` value in the request header, the server sends HTTP response code 428. If the `ETag` value is not correct, the server sends HTTP response code 412 .

You can update the name of the thing group, add new things or delete the already existing things, add new descriptions in supported languages, and change the existing descriptions. The valid operations are `ADD` and `DELETE`.

### Request

**URI:** `/v1/ThingGroups ('<thing group ID>')`

**Operation Type:** `CRUD`

**HTTP Method:** `PATCH`

**Permissions:** `<tg>.u`

### Request Header Parameters

Parameter	Required	Values
<code>Accept-Language</code>	No	Language of the thing type description. The default language is <code>en</code> .
<code>If-Match</code>	Yes	Latest <code>ETag</code> value of a specific thing group that is updated using the <code>PATCH</code> request.

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
<code>thing group ID</code>	Yes	String	Unique identifier of the thing group	Path

### Request Example

`/ThingGroup/v1/ThingGroups ('ABCGroup')`

### Payload

**Format:** `JSON`

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "Name": "ABCCarGroup",
  "ExternalID": "ABCCarGroup",
  "ThingGroupElements": [{
    "ThingID": "9FA169A380FC4A959B234D717E9C50A0",
    "Operation": "ADD"
  }, {
    "ThingID": "908758D995094A6F9CC72FC9C18BC366",
    "Operation": "DELETE"
  }],
  "ThingGroupDescriptions": [{
    "LanguageCode": "en",
    "Description": "Group of ABC Car Things"
  }, {
    "LanguageCode": "de",
    "Description": "Gruppe von ABC Car Things"
  }]
}
```

## Response

### Response Status and Error Codes

Code	Description
204	Thing group updated successfully

## Related Information

[Thing Group \[page 802\]](#)

## 12.5.2 Thing Group Element

Represents one or more elements of an individual thing group

**Resource Path:** `http://<server address>[:<port number>]/ThingGroup/v1/ThingGroupElements`

## Operations

### CRUD Operations

HTTP Method	Operation	URI
<i>GET</i>	<a href="#">Read Thing Group Elements [page 820]</a>	/v1/ThingGroupElements

### Thing Group Element Properties

Parameter	Data Type	Maximum Length	Description
GroupID	String	32	Unique identifier of the thing group generated by the system
ThingID	String	32	Unique identifier of the thing generated by the system
ThingName	String	255	Name of the thing
ThingDescription	String	255	Description of the thing
ThingAlternateID	String	255	Thing identifier which is unique only within a specific tenant. You cannot change the alternate thing ID.

### Related Information

[Grouping of Things \[page 801\]](#)

[Read Thing Group Elements \[page 820\]](#)

## 12.5.2.1 Read Thing Group Elements

With this service, you can retrieve all thing group elements.

### Request

**URI:** /v1/ThingGroupElements

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <tg>.r

### Request Example

```
/v1/ThingGroupElements
```



## Response

### Response Status and Error Codes

Code	Description
200	Thing group elements retrieved successfully

## Payload

### Format *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://apiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroupElements(GroupID='3E930040F5E3F8021600B10200CB288F',ThingID='503B8711B5
D0445688E7FF7ACFC2177A')",
          "uri": "https://apiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroupElements(GroupID='3E930040F5E3F8021600B10200CB288F',ThingID='503B8711B5
D0445688E7FF7ACFC2177A')",
          "type": "com.sap.apiot.ThingGroupElement"
        },
        "GroupID": "3E930040F5E3F8021600B10200CB288F",
        "ThingID": "503B8711B5D0445688E7FF7ACFC2177A",
        "ThingAlternateID": null,
        "ThingName": "ABCXSeries",
        "ThingDescription": "ABC X Series",
        "Operation": null
      },
      {
        "__metadata": {
          "id": "https://apiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroupElements(GroupID='3E930040F5E3F8021600B10200CB288F',ThingID='908758D995
094A6F9CC72FC9C18BC366')",
          "uri": "https://apiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroupElements(GroupID='3E930040F5E3F8021600B10200CB288F',ThingID='908758D995
094A6F9CC72FC9C18BC366')",
          "type": "com.sap.apiot.ThingGroupElement"
        },
        "GroupID": "3E930040F5E3F8021600B10200CB288F",
        "ThingID": "908758D995094A6F9CC72FC9C18BC366",
        "ThingAlternateID": null,
        "ThingName": "ABCZSeries",
        "ThingDescription": "ABC Z Series",
        "Operation": null
      },
      {
        "__metadata": {
          "id": "https://apiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroupElements(GroupID='45930040F5E3F8021600B10200CB288F',ThingID='22BCCAFA99
7F4C94BE337962A9FE6FFE')",
          "uri": "https://apiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroupElements(GroupID='45930040F5E3F8021600B10200CB288F',ThingID='22BCCAFA99
7F4C94BE337962A9FE6FFE')",
          "type": "com.sap.apiot.ThingGroupElement"
        },
        "GroupID": "45930040F5E3F8021600B10200CB288F",
        "ThingID": "22BCCAFA997F4C94BE337962A9FE6FFE",
        "ThingAlternateID": null,
        "ThingName": "ABCSeries",
        "ThingDescription": "ABC Series",
        "Operation": null
      }
    ]
  }
}
```

```

        "uri": "https://appiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroupElements (GroupID='45930040F5E3F8021600B10200CB288F',ThingID='22BCCAFA99
7F4C94BE337962A9FE6FFE')",
        "type": "com.sap.appiot.ThingGroupElement"
    },
    {
        "GroupID": "45930040F5E3F8021600B10200CB288F",
        "ThingID": "22BCCAFA997F4C94BE337962A9FE6FFE",
        "ThingAlternateID": null,
        "ThingName": "ABC2Series",
        "ThingDescription": "ABC 2 Series",
        "Operation": null
    },
    {
        "__metadata": {
            "id": "https://appiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroupElements (GroupID='45930040F5E3F8021600B10200CB288F',ThingID='282BD6F44C
994307A910066FB50203A8')",
            "uri": "https://appiot-thing-group-
ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/
ThingGroupElements (GroupID='45930040F5E3F8021600B10200CB288F',ThingID='282BD6F44C
994307A910066FB50203A8')",
            "type": "com.sap.appiot.ThingGroupElement"
        },
        "GroupID": "45930040F5E3F8021600B10200CB288F",
        "ThingID": "282BD6F44C994307A910066FB50203A8",
        "ThingAlternateID": null,
        "ThingName": "ABC2sSeries",
        "ThingDescription": "ABC 2s Series",
        "Operation": null
    }
]
}
}

```

## Related Information

[Thing Group Element \[page 819\]](#)

## 12.5.3 Thing Group Description

Represents a thing group description

### Request

**URI:** `http://<server address>[:<port number>]/ThingGroup/v1/ThingGroups('<thing ID>')?
$expand=ThingGroupDescriptions`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

Permissions: <tg>.r

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
GroupID	Yes	String	Unique identifier of the thing group	Path

## Response

### Response Status and Error Codes

Code	Description
200	Thing group description retrieved successfully

## Payload

### Format *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure

/ThingGroup/v1/ThingGroups('3E930040F5E3F8021600B10200CB288F')?  
\$expand=ThingGroupDescriptions

```
{
  "d": {
    "__metadata": {
      "id": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroups('3E930040F5E3F8021600B10200CB288F')",
      "uri": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroups('3E930040F5E3F8021600B10200CB288F')",
      "type": "com.sap.appiot.ThingGroup"
    },
    "GroupID": "3E930040F5E3F8021600B10200CB288F",
    "Name": "ABCCarGroup",
    "Description": null,
    "ExternalID": "ABCCarGroup",
    "ThingGroupElements": {
      "__deferred": {
        "uri": "https://appiot-thing-group-ci.cfapps.sap.hana.ondemand.com:443/ThingGroup/v1/ThingGroups('3E930040F5E3F8021600B10200CB288F')/ThingGroupElements"
      }
    },
    "ThingGroupDescriptions": {
      "results": [
        {
          "LanguageCode": "en",
          "Description": "Group of ABC Car Things"
        },
        {
          "LanguageCode": "de",
          "Description": "Gruppe von ABC Car Things"
        }
      ]
    }
  }
}
```

```
}  
}  
}
```

## Related Information

[Thing Group \[page 802\]](#)

# 12.6 Association of Things

## Overview

You can associate one or more thing instances to another thing instance. This is only possible if an association already exists between the thing types of these thing instances. For example, the thing type **Truck** has an association with the thing type **Car** and the relationship is `HAS_A`. Based on this thing type association, you can associate the thing instances of thing types **Truck** and **Car**.

### OData Version: 2.0

**Root URI:** `http://<server address>[:<port number>][/path]/ThingAssociation`

**Example for a base URI in a cloud foundry environment:** `https://thing-associations-sap.cfapps.eu10.hana.ondemand.com/ThingAssociation`

**Service Metadata URI:** `https://thing-associations-sap.cfapps.eu10.hana.ondemand.com/ThingAssociation/v1/$metadata`

## Operations

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read Metadata of Thing Association [page 826]</a>	<code>/ThingAssociation/v1/\$metadata</code>	<code>&lt;thing&gt;.r</code>
POST	<a href="#">Create Thing Association [page 829]</a>	<code>/ThingAssociation/v1/Thing('&lt;thing ID&gt;')/Associations</code>	<code>&lt;thing&gt;.c</code>
GET	<a href="#">Read All Associations of a Thing [page 831]</a>	<code>/ThingAssociation/v1/Thing('&lt;thing ID&gt;')/Associations</code>	<code>&lt;thing&gt;.r</code>

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read Details of a Thing Association [page 834]</a>	/ThingAssociation/v1/ Associations ('<association ID>')	<thing>.r
DELETE	<a href="#">Delete Association of a Thing [page 836]</a>	/ThingAssociation/v1/ Associations ('<association ID>')	<thing>.d

## Thing Association Object Properties

Property	Type	Mandatory	Maximum Length	Description
AssociatedThingId	String	Yes	32	Unique identifier of the associated thing
Description	String	No	60	Description of the thing association for each language with the corresponding ISO language code
Relationship	String	Yes	50	Type of relationship between the primary thing and the associated thing
ExpiryTime	TimeStamp	No	50	Time after which the association between the things becomes invalid
PrimaryThingID	String	No	60	Unique identifier of the primary thing to which other things are associated
CreationTime	TimeStamp	No	60	Time when the association is created, which is the current time
AssociationId	String	No	32	Unique identifier of the thing association

## Thing Association Description Properties

Property	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the object description

Property	Data Type	Maximum Length	Description
Description	String	255	Description of the object

## 12.6.1 Read Metadata of Thing Association

With this service, you can retrieve the metadata of thing association.

### Request

**URI:** `/ThingAssociation/v1/$metadata`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thing>.r`

### Request Example

`/ThingAssociation/v1/$metadata`

### Response

#### Response Status and Error Codes

Code	Description
200	Metadata of thing association retrieved successfully

### Payload

**Format:** [XML](#)

```
<?xml version="1.0" ?>
<edmx:Edmx Version="1.0" xmlns:edmx="http://schemas.microsoft.com/ado/2007/06/edmx">
  <edmx:DataServices m:DataServiceVersion="1.0" xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:sap="http://www.sap.com/Protocols/SAPData">
    <Schema Namespace="com.sap.apptot" xmlns="http://schemas.microsoft.com/ado/2008/09/edm">
      <EntityType Name="Thing">
        <Documentation>
          <Summary>Things which the user can have access to</Summary>
        </Documentation>
      </EntityType>
    </Schema>
  </DataServices>
</edmx:Edmx>
```

```

        <Key>
          <PropertyRef Name="ThingId"></PropertyRef>
        </Key>
        <Property Name="ThingId" Type="Edm.String" Nullable="false"
MaxLength="60" sap:creatable="true" sap:updatable="false">
          <Documentation>
            <Summary>Details of the Thing Id</Summary>
          </Documentation>
        </Property>
        <NavigationProperty Name="Associations"
Relationship="com.sap.apiot.ThingsToThingAssociation" FromRole="Thing"
ToRole="Association"></NavigationProperty>
      </EntityType>
      <EntityType Name="Association">
        <Documentation>
          <Summary>Details of Association type entity</Summary>
        </Documentation>
        <Key>
          <PropertyRef Name="AssociationId"></PropertyRef>
        </Key>
        <Property Name="AssociationId" Type="Edm.String"
Nullable="false" MaxLength="60" sap:creatable="true" sap:updatable="false">
          <Documentation>
            <Summary>Details of the AssociationId</Summary>
          </Documentation>
        </Property>
        <Property Name="PrimaryThingId" Type="Edm.String"
Nullable="true" MaxLength="60" sap:creatable="true" sap:updatable="true">
          <Documentation>
            <Summary>Primary Thing Id. This is the primary entity in
the association</Summary>
          </Documentation>
        </Property>
        <Property Name="AssociatedThingId" Type="Edm.String"
Nullable="true" MaxLength="60" sap:creatable="true" sap:updatable="true">
          <Documentation>
            <Summary>Associated Thing Id. This is the secondary
entity in the association</Summary>
          </Documentation>
        </Property>
        <Property Name="Relationship" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="true">
          <Documentation>
            <Summary>Relation type of the things (Ex: IS-A, HAS-A,
USES-A, IS-PART-OF)</Summary>
          </Documentation>
        </Property>
        <Property Name="ExpiryTime" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="true">
          <Documentation>
            <Summary>Expiry time of the association between things</
Summary>
          </Documentation>
        </Property>
        <Property Name="CreationTime" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="false" sap:updatable="true">
          <Documentation>
            <Summary>Creation time of the association</Summary>
          </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="true" sap:updatable="true">
          <Documentation>
            <Summary>Description of package entity</Summary>
          </Documentation>
        </Property>

```

```

        <NavigationProperty Name="Descriptions"
Relationship="com.sap.appiot.ThingAssociationToDescriptionAssociation"
FromRole="ThingAssociation" ToRole="Description"></NavigationProperty>
    </EntityType>
    <EntityType Name="Description">
        <Documentation>
            <Summary>Details of description type entity</Summary>
        </Documentation>
        <Key>
            <PropertyRef Name="LanguageCode"></PropertyRef>
        </Key>
        <Property Name="LanguageCode" Type="Edm.String" Nullable="false"
MaxLength="2" sap:creatable="true" sap:updatable="false">
            <Documentation>
                <Summary>Language code of description</Summary>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="60" sap:creatable="true" sap:updatable="true">
            <Documentation>
                <Summary>Description of package entity</Summary>
                <LongDescription>Long description of description entity</
LongDescription>
            </Documentation>
        </Property>
    </EntityType>
    <Association Name="ThingsToThingAssociation">
        <End Type="com.sap.appiot.Thing" Multiplicity="1" Role="Thing"></
End>
        <End Type="com.sap.appiot.Association" Multiplicity="*"
Role="Association"></End>
    </Association>
    <Association Name="ThingAssociationToDescriptionAssociation">
        <End Type="com.sap.appiot.Association" Multiplicity="1"
Role="ThingAssociation"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <EntityContainer Name="ThingAssociationContainer"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="Things" EntityType="com.sap.appiot.Thing">
            <Documentation>
                <Summary>Things which the user can have access to</
Summary>
            </Documentation>
        </EntitySet>
        <EntitySet Name="Associations"
EntityType="com.sap.appiot.Association">
            <Documentation>
                <Summary>Details of association set type</Summary>
            </Documentation>
        </EntitySet>
        <EntitySet Name="Descriptions"
EntityType="com.sap.appiot.Description">
            <Documentation>
                <Summary>Details of description set type</Summary>
            </Documentation>
        </EntitySet>
        <AssociationSet Name="ThingsToThingAssociation"
Association="com.sap.appiot.ThingsToThingAssociation">
            <End EntitySet="Things" Role="Thing"></End>
            <End EntitySet="Associations" Role="Association"></End>
        </AssociationSet>
        <AssociationSet Name="ThingAssociationToDescriptionAssociation"
Association="com.sap.appiot.ThingAssociationToDescriptionAssociation">
            <End EntitySet="Associations" Role="ThingAssociation"></End>
            <End EntitySet="Descriptions" Role="Description"></End>
        </AssociationSet>

```



```
</EntityContainer>
</Schema>
</edmx:DataServices>
</edmx:Edmx>
```

## 12.6.2 Create Thing Association

With this service, you can create association between valid thing instances.

The expiry time of the thing association must be greater than the current date and less than the expiry time defined for the corresponding thing type association.

### Note

You can associate any number of things to the primary thing, if the corresponding thing types are already associated. However, you can associate only one thing with the primary thing per `POST` request. You can associate more things with the primary thing through multiple `POST` requests.

## Request

**URI:** `/ThingAssociation/v1/Thing('<thing ID>')/Associations`

**Operation Type:** CRUD

**HTTP Method:** `POST`

**Permissions:** `<thing>.c`

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
thingID	Yes	String	Unique identifier of the thing	Path

### Request Example

`/ThingAssociation/v1/Thing('B36C9F7192954DA9A46AD50DDA521B13')/Associations`

## Payload

**Format:** `JSON`

```
{
  "AssociatedThingId": "4B90908372BB4648BEA1EFDD56B0819C",
  "Descriptions": [
    {
      "languageCode": "en",
      "description": "association between Truck thing and ABCXSeries thing"
    }
  ]
}
```

```

    }
  ],
  "Relationship": "HAS A",
  "ExpiryTime": "2018-04-16T08:26:56.229Z"
}

```

## Response

### Response Status and Error Codes

Code	Description
201	Thing association created successfully

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as illustrated in the following examples.

The following is the response payload after creating the thing type association.

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://thing-associations-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/Associations('CF42E985C3104789BD128CBC8A6B996B')",
          "uri": "https://thing-associations-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/Associations('CF42E985C3104789BD128CBC8A6B996B')",
          "type": "com.sap.apptot.Association"
        },
        "AssociationId": "CF42E985C3104789BD128CBC8A6B996B",
        "PrimaryThingId": "B36C9F7192954DA9A46AD50DDA521B13",
        "AssociatedThingId": "4B90908372BB4648BEA1EFDD56B0819C",
        "Relationship": "HAS A",
        "ExpiryTime": "2018-04-16T08:26:56.229Z",
        "CreationTime": "2017-12-12T10:31:12.581Z",
        "Description": "association between Truck thing and ABCXSeries
thing",
        "Descriptions": {
          "__deferred": {
            "uri": "https://thing-associations-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/Associations('CF42E985C3104789BD128CBC8A6B996B')/Descriptions"
          }
        }
      }
    ]
  }
}

```

## Related Information

[Association of Things \[page 824\]](#)

### 12.6.3 Read All Associations of a Thing

With this service, you can retrieve all the associated things for the specified thing.

#### Request

**URI:** `/ThingAssociation/v1/Thing('<thing ID>')/Associations`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<thing>.r`

#### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
thingID	Yes	String	Unique identifier of the thing	Path

#### Query String Parameters

Parameter	Required	Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$skip	No	Integer	Number of the first n records to be excluded from the result set.

Parameter	Required	Type	Description
\$inlinecount	No	String	<p>Total number of thing types available; permissible values are <i>allpages</i> and <i>none</i>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"> <li>• If you use this with the \$stop and \$skip parameters, the result contains the total count of thing types available.</li> <li>• If you use this with \$filter parameter, the result contains the total count of thing types based on the defined filter condition.</li> </ul>
\$filter	No	String	<p>Filter condition to be applied; the only valid fields are AssociatedThingId, Relationship, and AssociationId. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the <i>AND</i> separator.</p> <p>You can use the string functions such as <i>substringof</i>, <i>startswith</i>, and <i>endswith</i> to search data using \$filter. Valid fields are AssociatedThingId, Relationship, and AssociationId.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are PrimaryThingId, AssociationId, AssociatedThingId, CreationTime, Relationship, and ExpiryTime.</p>

Parameter	Required	Type	Description
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; valid fields are <code>AssociatedThingId</code> , <code>AssociationId</code> , and <code>Relationship</code> .  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.

## Request Example

```
/ThingAssociation/v1/Thing('B36C9F7192954DA9A46AD50DDA521B13')/Associations
```

## Response

### Response Status and Error Codes

Code	Description
200	Associated things of the specified thing retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as illustrated in the following examples.

```
/ThingAssociation/v1/Thing('B36C9F7192954DA9A46AD50DDA521B13')/Associations
```

Retrieves all the associated things of the thing B36C9F7192954DA9A46AD50DDA521B13.

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://thing-associations-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/Associations('F31184188A294A2E8BD769564DC4E049')",
          "uri": "https://thing-associations-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/Associations('F31184188A294A2E8BD769564DC4E049')",
          "type": "com.sap.apptot.Association"
        },
        "AssociationId": "F31184188A294A2E8BD769564DC4E049",
        "PrimaryThingId": "B36C9F7192954DA9A46AD50DDA521B13",
        "AssociatedThingId": "4B90908372BB4648BEA1EFDD56B0819C",
      }
    ]
  }
}
```

```

        "Relationship": "IS A",
        "ExpiryTime": "2018-10-16T08:26:56.000Z",
        "CreationTime": "2017-12-12T10:23:57.000Z",
        "Description": "association between Truck thing and ABC2Series
thing",
        "Descriptions": {
            "__deferred": {
                "uri": "https://thing-associations-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/
Associations('F31184188A294A2E8BD769564DC4E049')/Descriptions"
            }
        },
        {
            "__metadata": {
                "id": "https://thing-associations-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/
Associations('CF42E985C3104789BD128CBC8A6B996B')",
                "uri": "https://thing-associations-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/
Associations('CF42E985C3104789BD128CBC8A6B996B')",
                "type": "com.sap.apptot.Association"
            },
            "AssociationId": "CF42E985C3104789BD128CBC8A6B996B",
            "PrimaryThingId": "B36C9F7192954DA9A46AD50DDA521B13",
            "AssociatedThingId": "4B90908372BB4648BEA1EFDD56B0819C",
            "Relationship": "HAS A",
            "ExpiryTime": "2018-04-16T08:26:56.229Z",
            "CreationTime": "2017-12-12T10:31:12.581Z",
            "Description": "association between Truck thing and ABCXSeries
thing",
            "Descriptions": {
                "__deferred": {
                    "uri": "https://thing-associations-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/
Associations('CF42E985C3104789BD128CBC8A6B996B')/Descriptions"
                }
            }
        }
    ]
}

```

## Related Information

[Association of Things \[page 824\]](#)

## 12.6.4 Read Details of a Thing Association

With this service, you can retrieve the details of a thing association with a specific association ID.

### Request

**URI:** /ThingAssociation/v1/Associations('<association ID>')

Operation Type: CRUD

HTTP Method: [GET](#)

Permissions: <thing>.r

## Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
AssociationId	Yes	String	Unique identifier of the thing association	Path

## Request Example

/ThingAssociation/v1/Associations('CF42E985C3104789BD128CBC8A6B996B')

## Response

### Response Status and Error Codes

Code	Description
200	Details of a specific thing association retrieved successfully

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

/ThingAssociation/v1/Associations('CF42E985C3104789BD128CBC8A6B996B')

Retrieves the association details with the association ID 'CF42E985C3104789BD128CBC8A6B996B'.

```
{
  "d": {
    "__metadata": {
      "id": "https://thing-associations-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/Associations('CF42E985C3104789BD128CBC8A6B996B')",
      "uri": "https://thing-associations-sap-ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/Associations('CF42E985C3104789BD128CBC8A6B996B')",
      "type": "com.sap.apptot.Association"
    },
    "AssociationId": "CF42E985C3104789BD128CBC8A6B996B",
    "PrimaryThingId": "B36C9F7192954DA9A46AD50DDA521B13",
    "AssociatedThingId": "4B90908372BB4648BEA1EFDD56B0819C",
    "Relationship": "HAS A",
    "ExpiryTime": "2018-04-16T08:26:56.229Z",
    "CreationTime": "2017-12-12T10:31:12.581Z",
    "Description": "association between Truck thing and ABCXSeries thing",
    "Descriptions": {
      "__deferred": {
```

```

        "uri": "https://thing-associations-sap-
ci.cfapps.sap.hana.ondemand.com:443/ThingAssociation/v1/
Associations('CF42E985C3104789BD128CBC8A6B996B')/Descriptions"
    }
}
}
}

```

## Related Information

[Association of Things \[page 824\]](#)

## 12.6.5 Delete Association of a Thing

With this service, you can delete a thing association with a specific association ID.

The `ETag` value is mandatory to delete a thing association. You must retrieve the `ETag` value of the thing association using the GET (`/ThingAssociation/v1/Associations('<association ID>')`) service. The `ETag` value is assigned to the request header parameter `If-Match` in the DELETE request. If you do not include the `ETag` value in the request header, the server sends HTTP response code 428. If the `ETag` value is not correct, the server sends HTTP response code 412.

### Request

**URI:** `/ThingAssociation/v1/Associations('<association ID>')`

**Operation Type:** CRUD

**HTTP Method:** *DELETE*

**Permissions:** `<thing>.d`

### Request Headers

Header	Required	Values
<code>If-Match</code>	Yes	Latest <code>ETag</code> value of a specific thing association.

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
<code>AssociationId</code>	Yes	String	Unique identifier of the thing association	Path

### Request Example

`/ThingAssociation/v1/Associations('CF42E985C3104789BD128CBC8A6B996B')`



## Response

### Response Status and Error Codes

Code	Description
204	Association between things with a specific association ID is deleted successfully

## Related Information

[Association of Things \[page 824\]](#)

## 12.7 Read Version Information of Thing Service

This service is used to retrieve the version information of the thing service.

## Request

URI: `/Version`

HTTP Method: `GET`

Permissions:

`<thing>.r`

### Request Example

```
/Version
```

## Response

Format: `JSON`

### Response Status and Error Codes

Code	Description
200	Version information retrieved successfully.

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  "version" : "1.10"
}
```

## 12.8 HTTP Status Codes for Thing Services

List of status codes for thing services

In the following table, you find the list of HTTP status codes that are used by the thing services:

Post (Create)	Post (Update)	Put	Delete	Get (single value)	Get (collection)	Description
201	200	200	204	200	200	Success
403	404	404	404	404	200	No capabilities assigned.
201	404	200	404	404	200	Read capability not assigned to read a specific thing. However, the user is assigned with the capabilities to perform operations such as create, update, and delete.

**i Note**

For the **Delete Time Series Data** service, the system displays 204 as HTTP status code.

Post (Create)	Post (Update)	Put	Delete	Get (single value)	Get (collection)	Description
403	404	404	404	200	200	Read capability assigned to read a specific thing. However, the user is not assigned with capabilities to perform operations such as create, update, and delete.
n/a	404	404	404	404	n/a	Not found  The requested thing is not found.
403	403	403	403	403	403	Forbidden (scope(s) not assigned for the user)
400	400	400	400	400	400	Bad request (for example: header field missing, wrong filter, validation failed)
500	500	500	500	500	500	Internal server error

# 13 Thing: Master Data and Parameters Data

Data specific to a thing

After creating a thing instance of a specific thing type, you can use the thing services to create, retrieve, and delete property data of things. The services explained in this section only support property data of data categories **MasterData** and **Parameters**.

## Base URI

- **Formal description:** `http://<server address>[:<port number>][/path]/Things`
- **Example for a base URI in a cloud foundry environment:** `https://apiot-mds.cfapps.eu10.hana.ondemand.com/Things`

## Methods

HTTP Method	Action	URI	Scopes
<i>PUT</i>	<a href="#">Create Data for a Thing [page 845]</a>	<code>/Things('&lt;thing ID&gt;')/&lt;thing type name&gt;/&lt;property set ID&gt;</code>	<code>&lt;thing&gt;.c</code>
<i>GET</i>	<a href="#">Read Data for a Thing [page 849]</a>	<code>/Things('&lt;thing ID&gt;')/&lt;thing type name&gt;/&lt;property set ID&gt;</code>	<code>&lt;thing&gt;.r</code>
<i>DELETE</i>	<a href="#">Delete Data for a Thing [page 853]</a>	<code>/URL/Things('&lt;thing ID&gt;') /&lt;property set ID&gt;?timerange=&lt;From time&gt;- &lt;To time&gt; (OR) &lt;Thing Application&gt;/URL/Things('&lt;thingID&gt;')/&lt;property set ID&gt;?timerange=&lt;1H, 1D, 1W, 1M&gt;</code>	<code>&lt;thing&gt;.d</code>
<i>GET</i>	<a href="#">Read Metadata for a Thing [page 840]</a>	<code>/Things('&lt;thing ID&gt;') / Configuration</code>	
<i>GET</i>	<a href="#">Read Property Set Data [page 854]</a>	<code>/Things/ PropertySet/&lt;property set type name&gt;</code>	<code>&lt;thing&gt;.r</code>

## 13.1 Read Metadata for a Thing

Retrieves metadata for a thing

With this service, you can retrieve metadata for a specified thing.

## Request

URI: `/Things('<ID>')/Configuration`

HTTP Method: [GET](#)

### Request Headers

Header	Required	Values
Accept-Language	No	Language of the thing type description. The default language is <a href="#">EN</a> .
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header <code>CONTENT-ENCODING</code> with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the un-compressed response.

### Request Parameter

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>

### Request Example

```
/Things('<ID>')/Configuration
```

## Response

Format: [JSON](#)

### Response Properties

Property	Type	Description
Value	Array of thing type objects	Array of thing type objects returned.
propertySets	Array of property set objects	Array of property set objects (that belong to the property set type) returned.

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  "thingTypes": [{
    "name": "core.automobiles:ABC2Series",
    "description": {
      "en": "ABC2Series electric car"
    },
    "propertyTypes": [{
      "id": "engine",
      "description": {
        "en": "Engine Usage"
      },
      "propertySetType": "core.automobiles:Engine"
    }, {
      "id": "drive_fr",
      "description": {
        "en": "Drive Front Right"
      },
      "propertySetType": "core.automobiles:DriveUnit"
    }, {
      "id": "drive_fl",
      "description": {
        "en": "Drive Front Left"
      },
      "propertySetType": "core.automobiles:DriveUnit"
    }, {
      "id": "car",
      "description": {
        "en": "Car Usage"
      },
      "propertySetType": "core.automobiles:Car"
    }, {
      "id": "wheel_fr",
      "description": {
        "en": "Wheel Front Right"
      },
      "propertySetType": "core.automobiles:Wheel"
    }, {
      "id": "wheel_fl",
      "description": {
        "en": "Wheel Front Left"
      },
      "propertySetType": "core.automobiles:Wheel"
    }, {
      "id": "wheel_rr",
      "description": {
        "en": "Wheel Rear Right"
      },
      "propertySetType": "core.automobiles:Wheel"
    }, {
      "id": "wheel_rl",
      "description": {
        "en": "Wheel Rear Left"
      },
      "propertySetType": "core.automobiles:Wheel"
    }
  ]
}, {
  "propertySetTypes": [{
    "name": "core.automobiles:Car",
    "description": {
      "en": "Master Data for car"
    }
  }]
}]
```

```

    },
    "dataCategory": "MasterData",
    "propertyTypes": [{
      "value": {
        "id": "BodyStyle",
        "description": {
          "en": "Car Body Style"
        },
        "type": "String",
        "length": "127"
      }
    }, {
      "value": {
        "id": "Class",
        "description": {
          "en": "Car Class"
        },
        "type": "String",
        "length": "127"
      }
    }, {
      "value": {
        "id": "WheelBase",
        "description": {
          "en": "Car Wheel Base"
        },
        "type": "Numeric",
        "length": ""
      }
    }, {
      "value": {
        "id": "RegistrationNumber",
        "description": {
          "en": "Car Registration Number"
        },
        "type": "String",
        "length": "127"
      }
    }, {
      "value": {
        "id": "Colour",
        "description": {
          "en": "Car Colour"
        },
        "type": "String",
        "length": "127"
      }
    }
  ]
}, {
  "name": "core.automobiles:Wheel",
  "description": {
    "en": "Sensory parameters for wheel"
  },
  "dataCategory": "TimeSeriesData",
  "propertyTypes": [{
    "value": {
      "id": "Pressure",
      "description": {
        "en": "Wheel pressure"
      },
      "type": "Numeric",
      "length": "",
      "unitOfMeasure": "PA"
    }
  }, {
    "value": {
      "id": "Temperature",
      "description": {

```

```

        "en": "Wheel temperature"
    },
    "type": "Numeric",
    "length": ""
}
    ]
}, {
    "name": "core.automobiles:DriveUnit",
    "description": {
        "en": "Sensory parameters for drive unit"
    },
    "dataCategory": "TimeSeriesData",
    "propertyTypes": [{
        "value": {
            "id": "RotationSpeed",
            "description": {
                "en": "Drive unit rotation speed"
            },
            "type": "Numeric",
            "length": ""
        }
    }, {
        "value": {
            "id": "PowerConsumption",
            "description": {
                "en": "Drive unit power consumption"
            },
            "type": "Numeric",
            "length": ""
        }
    }
    ],
}, {
    "name": "core.automobiles:Engine",
    "description": {
        "en": "Sensory parameters for engine"
    },
    "dataCategory": "MasterData",
    "propertyTypes": [{
        "value": {
            "id": "Type",
            "description": {
                "en": "Engine type"
            },
            "type": "String",
            "length": "127"
        }
    }, {
        "value": {
            "id": "Capacity",
            "description": {
                "en": "Engine capacity"
            },
            "type": "Numeric",
            "length": "",
            "unitOfMeasure": "MM"
        }
    }, {
        "value": {
            "id": "Torque",
            "description": {
                "en": "Engine torque"
            },
            "type": "Numeric",
            "length": "",
            "unitOfMeasure": "NM"
        }
    }, {
        "value": {

```



```

        "id": "Transmission",
        "description": {
            "en": "Engine transmission"
        },
        "type": "String",
        "length": "254",
        "unitOfMeasure": ""
    }
}
}
}
}

```

## 13.2 Create Data for a Thing

With this service, you can create data for the property sets of a thing that belongs to a specific thing type. This service supports data creation for the property sets of data categories **MasterData** and **Parameters**.

Using this service, you can send the request payload in *JSON* format or as a binary file in *zip* or *gzip* format. You can use the binary file format when the request payload is large. The request payload in binary file format saves space and reduces the transfer time as its size is reduced after compression. The complete request payload must be compressed in a single binary file and you cannot split the request payload in multiple files.

### Multiple Language Support for Data

You can create data for properties of data type *String* in multiple languages. To achieve this, when you create a property set type, you must define a field `IsMultiLingual=true` in the request payload.

## Request

**URI:** `/Things('<ID>')/<thing type name>/<property set ID>`

### Note

- The thing type must be a fully qualified name that is prefixed with the package name.
- The value of the *\_time* property must correspond to UTC timestamp. If you do not specify the *\_time* property, the system uses `utcnow()` value for the *\_time* property, by default. However, if you are creating multiple records of data, the *\_time* property is mandatory. Following are the supported formats:
  - `\Date(<ticks>)\<ticks>` = number of milliseconds since midnight Jan 1, 1970"  
For example, `"\Date(1546322403148)/"`
  - `"YYYY-MM-DD HH:MM:SS[.fraction]"`  
For example, `"2015-08-01T09:00:00.000Z"`

**HTTP Method:** *PUT*

**Permissions:** `<thing>.c`

## Request Header Parameters

The following request header parameter is applicable only when you want to send the request payload in binary file format.

Header	Required	Values
content-Encoding	No	<a href="#">application/zip</a> or <a href="#">application/gzip</a>

## Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thing type name	Yes	String	Name of a thing type.
property set ID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

## Request Example

```
/Things('<thing ID>')/core.automobiles:ABCXSeries/car
```

## Response

Format: [JSON](#)

## Response Status and Error Codes

Code	Description
200	Data of category <b>MasterData</b> or <b>Parameters</b> for the specified thing created successfully

## Payload

Format: [JSON](#)

Only media type [JSON](#) is supported. In addition, you can encode the request payload in [JSON](#) format to [zip](#) or [gzip](#) file format using the request header parameter `content-Encoding`. The [JSON](#) for this method has the structure as illustrated in the following examples:

### Example 1

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/core.automobiles:ABCXSeries/car
```

The following request payload is used to create data for the property set `car`.

```
{
  "value": [
    {
      "_time": "2015-08-01T09:00:00Z",
      "BodyStyle": "Sedan",
      "Class": "1st",
      "WheelBase": "10",
      "RegistrationNumber": "007",
      "Colour": "Red"
    },
    {
      "_time": "2015-08-02T06:00:03Z",
      "BodyStyle": "SUV",
      "Class": "1st",
      "WheelBase": "15",
      "RegistrationNumber": "001",
      "Colour": "Black"
    }
  ]
}
```

## Example 2

/Things('503B8711B5D0445688E7FF7ACFC2177A')/core.automobiles:ABCXSeries/carstyle

The following request payload is used to create data in multiple languages for the property `Color` of the property set `carstyle`.

### Note

- If you do not specify the language code, the system considers the Locale specific language as the default language.
- Only the *JSON* format is supported to specify the data in multiple languages. If the data is not in *JSON* format, the system saves the data as a string in the *Locale* specific language for the property.
- In the following example, the string property `LicenseNumber` supports multiple languages but the data is not in *JSON* format. Hence, the data is saved as a string in the *Locale* specific language.

```
{
  "value": [{
    "_time": "2015-08-01T09:00:00Z",
    "Color": "{\"en\":\"Blue\",\"de\":\"Blau\"}",
    "Model": "ABC X Series Model",
    "ModelNumber": "10",
    "LicenseNumber": "ABC10"
  },
  {
    "_time": "2015-08-02T06:00:03Z",
    "Color": "{\"en\":\"White\",\"de\":\"Weiß\"}",
    "Model": "ABC X Series Model",
    "ModelNumber": "15",
    "LicenseNumber": "ABC15"
  },
  {
    "_time": "2015-09-02T06:00:03Z",
    "Color": "{\"en\":\"White\",\"de\":\"Weiß\"}",
    "Model": "ABC X Series Model",
    "ModelNumber": "25",
    "LicenseNumber": "ABC25"
  }
]
```

## 13.3 Create Data Using Alternate Thing ID

With this service, you can create data for the property sets of a thing that belongs to a specific thing type. You must provide the alternate thing ID in the request URL. This service supports data creation for the property sets of data categories *MasterData* and *Parameters*.

### Request

**URI:** `/ThingsByAlternateId('<alternate thing ID>')/<thing type name>/<property set ID>`

**HTTP Method:** *PUT*

**Permissions:** `<thing>.c`

### Request Parameters

Parameter	Required	Data Type	Description
alternate thing ID	Yes	String	Alternate thing identifier unique within a specific tenant.
thing type name	Yes	String	Name of a thing type.
property set ID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

### Request Example

`/ThingsByAlternateId('ABCXSeries001-alternateId')/core.automobiles:ABCXSeries/car`

### Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  "value": [
    {
      "time": "2015-08-01T09:00:00Z",
      "BodyStyle": "Sedan",
      "Class": "1st",
      "WheelBase": "10",
      "WheelBase_QC": "8",
      "RegistrationNumber": "007",
      "Colour": "Red"
    },
    {
      "time": "2015-08-02T06:00:03Z",
      "BodyStyle": "SUV",
      "Class": "1st",
```

```

        "WheelBase": "15",
        "WheelBase_QC": "9",
        "RegistrationNumber" : "001",
        "Colour": "Black"
    }
}

```

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
200	Data of category <b>MasterData</b> or <b>Parameters</b> for the specified thing created successfully

## Related Information

[Create Data for a Thing \[page 845\]](#)

## 13.4 Read Data for a Thing

Retrieves property set data for the specified thing

With this service, you can read the values of property sets for the specified thing type. You can request the values for a specific time range. The service retrieves the values of property sets of data categories **MasterData** or **Parameters**.

### Note

Retrieving number of objects more than the recommended limits with only one collective GET request can lead to high system load and a significant increase of response times.

To avoid performance decline, the system in future versions will automatically limit collective GET requests to a maximum of 500 objects that can be retrieved with one single call.

Use of `$count` as well degrades performance. Hence, a use of `$count` must also be judiciously chosen.

Therefore, for use cases where it is crucial for you to ensure that objects more than 500 have to be retrieved, you need to use paging options such as `$top` and `$skip`.

## Request

URI: /Things('<ID>')/<thing type name>/<property set ID>

### Note

- The thing type has to be a fully qualified name that is prefixed with the package.
- If you do not specify a time range, the data that is available for the latest time value is retrieved.

HTTP Method: [GET](#)

Permissions:

<thing>.r

## Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header CONTENT-ENCODING with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Parameter

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thing type name	Yes	string	Name of the thing type.
property set ID	Yes	string	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

## Query String Parameters

Parameter	Required	Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$skip	No	Integer	Number of the first n records to be excluded from the result set.

Parameter	Required	Type	Description
\$count	No	Boolean	<p>Total number of time series objects available based on the current user's authorizations; permissible values are <i>true</i> and <i>false</i>.</p> <p>The following results apply, if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"> <li>• If you use this with the \$top and \$skip parameters, the result contains the total count of time series objects available.</li> <li>• If you use this with \$filter parameter, the result contains the total count of time series objects based on the defined filter condition.</li> </ul>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; valid fields are <i>_time</i> and all the property types.</p> <div> <p><b>i Note</b></p> <p>Data types <i>JSON</i> and <i>ByteArray</i> are not supported.</p> </div> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; valid fields are <i>_time</i> and all the property types. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <div> <p><b>i Note</b></p> <p>Data types <i>JSON</i> and <i>ByteArray</i> are not supported.</p> </div>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are <i>_time</i> and all the property types.</p>

## Request Example

```
/Things('<ID>')/core.automobiles:ABCXSeries/car?timerange=10M
```

Retrieves the property set type car for the thing ID `core.automobiles:ABCXSeries` corresponding to a time range of 10 months.

```
/Things('<ID>')/core.automobiles:ABCXSeries/car?timerange=10M&
$select=_time,BodyStyle
```

Retrieves the property set type car for the thing ID `core.automobiles:ABCXSeries` corresponding to a time range of 10 months. However, with `$select` query parameter, only the fields `_time` and `BodyStyle` are displayed for the retrieved thing.

```
/Things('<ID>')/core.automobiles:ABCXSeries/car?  
timerange=2015-08-01T09:00:00Z-2015-10-01T09:00:00Z
```

Retrieves the property set type car for the thing ID `core.automobiles:ABCXSeries` corresponding to a time range from August 1, 2015 to October 1, 2015.

## Response

Format: [JSON](#)

### Response Properties

Property	Type	Description
value	Array of thing data objects	An array of data objects (for example, time series data) for a thing or property set type.

### Response Status and Error Codes

Code	Description
200	Data for one or more things retrieved successfully

## Payload

Format: [JSON](#)

Only media type [JSON](#) is supported. The [JSON](#) for this method has the structure as illustrated in the following examples:

### Example 1

Retrieves data for the specified property set of a thing

```
{  
  "value": [  
    {  
      "BodyStyle": "Sedan",  
      "Class": "1st",  
      "WheelBase": "10",  
      "RegistrationNumber": "007",  
      "Colour": "Red",  
      "_time": "2015-08-01T09:00:00.000Z"  
    },  
    {  
      "BodyStyle": "SUV",  
      "Class": "1st",  
      "WheelBase": "15",  
    }  
  ]  
}
```



```

        "RegistrationNumber" : "001",
        "Colour": "Black",
        "_time": "2015-08-02T06:00:03.000Z"
    }
}

```

## Example 2

Retrieves data for the specified property set of a thing. Some of the string properties in this example supports multiple languages.

```

{
  "value": [{
    "_time": "2015-08-01T09:00:00Z",
    "Color": "{ \"en\": \"Blue\", \"de\": \"Blau\" }",
    "Model": "ABC X Series Model",
    "ModelNumber": "10",
    "LicenseNumber": "{ \"en\": \"ABC10\" }"
  },
  {
    "_time": "2015-08-02T06:00:03Z",
    "Color": "{ \"en\": \"White\", \"de\": \"Weiß\" }",
    "Model": "ABC X Series Model",
    "ModelNumber": "15",
    "LicenseNumber": "{ \"en\": \"ABC10\" }"
  },
  {
    "_time": "2015-09-02T06:00:03Z",
    "Color": "{ \"en\": \"White\", \"de\": \"Weiß\" }",
    "Model": "ABC X Series Model",
    "ModelNumber": "25",
    "LicenseNumber": "{ \"en\": \"ABC10\" }"
  }
]
}

```

## 13.5 Delete Data for a Thing

Deletes master data or parameters data for a thing

With this service, you can delete the master data or parameters data for the specified thing and property set type corresponding to the time range provided.

### Request

**URI:** /Things('<ID>')/<thing type name>/<property set ID>?timerange=<From time>-<To time> (OR) /Things('<ID>')/<thing type name>/<property set ID>? timerange=<1H, 1D, 1W, 1M>

#### i Note

You must specify a time range to be able to delete the master data or parameters data.

HTTP Method: [DELETE](#)

Permissions: `<thing>.d`

### Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
property set ID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

### Request Example

```
/Things('<thingID>')/core.automobiles:ABCXSeries/wheel_fl?  
timerange=2015-08-01T06:00:00Z-2015-08-02T09:00:00Z
```

## Response

### Response Status and Error Codes

Format: [JSON](#)

Code	Description
204	Data deleted successfully

## 13.6 Read Property Set Data

This service is used to return the things with the most recent property data for the specified property set type. This service only retrieves property set type data of data categories [MasterData](#) and [Parameters](#).

### Note

Retrieving number of objects more than the recommended limits with only one collective GET request can lead to high system load and a significant increase of response times.

To avoid performance decline, the system in future versions will automatically limit collective GET requests to a maximum of 500 objects that can be retrieved with one single call.

Use of `$count` as well degrades performance. Hence, a use of `$count` must also be judiciously chosen.

Therefore, for use cases where it is crucial for you to ensure that objects more than 500 have to be retrieved, you need to use paging options such as `$top` and `$skip`.

## Request

URI: /Things/PropertySet/<property set type name>

HTTP Method: [GET](#)

Permissions:

<thing>.r

## Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header CONTENT-ENCODING with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Paramter

Parameter	Required	Data Type	Description
name	Yes	string	Name of the property set type.

## Query String Parameters

Parameter	Required	Type	Description
\$count	No	Boolean	Total number of thing objects available for the specified property set type based on the current user's authorizations; permissible values are <a href="#">true</a> and <a href="#">false</a>  The following results apply, if this parameter is used in conjunction with the other query parameters: <ul style="list-style-type: none"><li>• If you use this with the \$top and \$skip parameters, the result contains the total count of thing objects available for the specified property set type</li><li>• If you use this with \$filter parameter, the result contains the total number of thing objects available for the specified property set type based on the defined filter condition.</li></ul>

Parameter	Required	Type	Description
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; the only valid field is <a href="#">_time</a> . This is required only if property set type is time-dependent.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$filter	No	String	Filter condition to be applied; valid fields are <a href="#">_thingId</a> , <a href="#">_time</a> and all properties. The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> .

## Request Example

```
/Things/PropertySet/core.automobiles:Car
```

## Response

Format: [JSON](#)

### Response Properties

Property	Type	Description
value	Array of thing data objects	An array of thing data objects containing the Thing ID and values of properties for the specified property set type.

### Response Status and Error Codes

Code	Description
200	Property set type data for a one or more things retrieved successfully

## Payload

Format: [JSON](#)

Only media type [JSON](#) is supported. The [JSON](#) for this method has the following structure:

```
{
  "value": [
    {
      "_id": "<thingID>/car",
      "_thingId": "<thingID>",
      "BodyStyle": "SUV",
      "Class": "1st",
      "WheelBase": "15",
      "RegistrationNumber" : "001",
      "Colour": "Black",

```

```
    "_time": "2015-08-02T06:00:03.000Z"  
  }  
]  
}
```

# 14 Thing: Reference Property Data

With this service, you can create, read, or delete data for the reference properties of a specific thing.

## Base URI

**Formal description:** `http://<server address>[:<port number>][<path>]`

**Example for a base URI in a cloud foundry environment:** `https://apiot-mds.cfapps.eu10.hana.ondemand.com/Things`

## Methods

HTTP Method	Action	URI	Scopes
<i>PUT</i>	<a href="#">Create Data for Reference Properties [page 859]</a>	For a single thing: <code>/Things('&lt;thingID&gt;') / ReferenceProperties/&lt;thing type name&gt;/&lt;property set ID&gt;</code>  For multiple things: <code>/Things/ ReferenceProperties/&lt;&lt;thing type name&gt;&gt;/&lt;&lt;property set ID&gt;&gt;</code>	<code>&lt;thing&gt;.c</code>
<i>GET</i>	<a href="#">Read Data for Reference Properties [page 862]</a>	<code>/Things('&lt;thingID&gt;') / ReferenceProperties/&lt;thing type name&gt;/&lt;property set ID&gt;</code>	<code>&lt;thing&gt;.r</code>
<i>GET</i>	<a href="#">Read Reference Properties Data for a List of Property Sets [page 864]</a>	<code>/Things('&lt;thingID&gt;')/&lt;thing type name&gt;/ ReferenceProperties/'&lt;property set ID1&gt;,&lt;property type ID1&gt;&amp;&lt;property set ID1&gt;,&lt;property type ID2&gt;&amp;&lt;property set ID2&gt;,&lt;property type ID2&gt;'?timerange=&lt;fromTime&gt;-&lt;toTime&gt;</code>	<code>&lt;thing&gt;.r</code>
<i>DELETE</i>	<a href="#">Delete Data for Reference Properties [page 868]</a>	<code>/Things('&lt;thingID&gt;') / ReferenceProperties/&lt;thing type name&gt;/&lt;property set ID&gt;</code>	<code>&lt;thing&gt;.d</code>

### i Note

`<thing>` refers to the thing application name.

## Related Information

[Reference Property Set Type \[page 331\]](#)

# 14.1 Create Data for Reference Properties

You can use this service to create data for reference properties of a specific property set that is assigned to a thing type. You can create data for a single thing or multiple things.

## Request

### URI:

For a single thing: `/Things('<ID>')/ReferenceProperties/<thing type name>/<property set ID>`

For multiple things: `/Things/ReferenceProperties/<thing type name>/<property set ID>`

### Note

- The value for the `_time` property in the request payload must correspond to UTC timestamp and must be less than or equal to `current time + 90 days`.
  - `"/Date(<ticks>)/"` <ticks> = number of milliseconds since midnight Jan 1, 1970  
For example, `"/Date(1546322403148)/"`
  - `"YYYY-MM-DD HH:MM:SS[.fraction]"`  
For example, `"2015-08-01T09:00:00.000Z"`
- To create data for reference properties of multiple things, you must provide the thing IDs in the request payload.
- A reference property of attribute type `com.sap.iot.core.TargetValue` is used to send desired value to the corresponding device. When you create data for this reference property, the system sends data only with the latest timestamp to the device. For example, assume the following:
  - Create data for the reference property with the timestamp **2015-08-01T09:10:00Z**.
  - In the subsequent call, create data for the same reference property with an earlier timestamp **2015-08-01T09:01:00Z**. The system only sends the data with the latest timestamp **2015-08-01T09:10:00Z** (created in the previous call) to the device.

**HTTP Method:** `PUT`

**Permissions:** `<thing>.c`

## Request Headers

Header	Required	Values
Accept-Language	No	Language of the reference data. The default language is en.
sap-iot-strong-validations	Yes	Validate the payload.  Only if this parameter is set true, the data types <i>String</i> , <i>Numeric</i> , <i>Boolean</i> , <i>ThingID</i> , <i>BPID</i> , <i>Timestamp</i> , <i>DateTime</i> , and <i>Date</i> , along with <i>JSON</i> , <i>ByteArray</i> invalid property names are validated.

## Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>Unique thing identifier generated by the system while creating a thing.</li><li>Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thing type name	Yes	string	Name of the thing type
property set ID	Yes	string	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

## Request Example

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/ReferenceProperties/  
core.automobiles:car/wheel_fl
```

Reference data is created for the property set **wheel\_fl** of the thing type **core.automobiles:car**

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

### Example 1

```
{  
  "value": [{  
    "time": "2015-08-01T09:00:00Z",  
    "PressureUnitOfMeasure": "P",  
    "RotationUnitOfMeasure": "A",  
    "PressureLowerThreshold": 10,  
    "RotationLowerThreshold": 15  
  }, {  
    "time": "2015-08-01T09:00:00Z",  
    "PressureUnitOfMeasure": "PP",  
    "RotationUnitOfMeasure": "AA",  
    "PressureLowerThreshold": 20,  
    "RotationLowerThreshold": 25  
  }]  
}
```



```
}
```

### Example 2

The following request payload creates data for the properties `RotationTargetValue` and `PressureTargetValue` of attribute `com.sap.iot.core.TargetValue`:

```
{
  "value": [{
    "_time": "2015-08-01T09:00:00Z",
    "PressureUnitOfMeasure": "P",
    "RotationUnitOfMeasure": "A",
    "PressureLowerThreshold": 10,
    "RotationLowerThreshold": 15,
    "RotationTargetValue": 200,
    "PressureTargetValue": 100
  }, {
    "_time": "2015-08-01T09:00:00Z",
    "PressureUnitOfMeasure": "PP",
    "RotationUnitOfMeasure": "AA",
    "PressureLowerThreshold": 20,
    "RotationLowerThreshold": 25,
    "RotationTargetValue": 2000,
    "PressureTargetValue": 1000
  }]
}
```

### Example 3

The following request payload creates data for the reference properties (`RotationTargetValue` and `PressureTargetValue` of attribute `com.sap.iot.core.TargetValue`) for multiple things:

`/Things/ReferenceProperties/core.automobiles:car/wheel_fl`

```
{
  "things": [{
    "id": "3E930040F5E3F8021600B10200CB288F"
  }, {
    "id": "45930040F5E3F8021600B10200CB288F"
  }],
  "value": [{
    "_time": "2015-08-01T09:00:00Z",
    "PressureUnitOfMeasure": "P",
    "RotationUnitOfMeasure": "A",
    "PressureLowerThreshold": 10,
    "RotationLowerThreshold": 15,
    "RotationTargetValue": 2000,
    "PressureTargetValue": 1000
  }]
}
```

## Response

Format: [JSON](#)

## Response Status and Error Codes

Code	Description
200	Reference data created successfully.

## Related Information

[Thing: Reference Property Data \[page 858\]](#)

## 14.2 Read Data for Reference Properties

You can use this service to read data of reference properties of a specific property set.

By default, the service retrieves data available for the last one hour. You can use `$filter` on `_time` property to retrieve data for a specific time or for a specific time range. The value for the `_time` property in the request URL must correspond to UTC timestamp.

## Request

**URI:** `/Things('<ID>')/ReferenceProperties/<thing type name>/<property set ID>`

**HTTP Method:** `GET`

**Permissions:** `<thing>.r`

## Request Header Parameters

Header	Required	Values
<code>Accept-Language</code>	No	Language of the reference data. The default language is en.
<code>Accept-Encoding</code>	No	Retrieve compressed response  Only if you specify the value <code>gzip</code> or <code>gzip,deflate</code> for this parameter, the system returns the compressed response with an additional response header <code>CONTENT-ENCODING</code> with value <code>gzip</code> . If you specify an invalid value or do not use this header parameter, the system returns the un-compressed response.

## Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thing type name	Yes	string	Name of the thing type
property set ID	Yes	string	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

## Query String Parameters

Parameter	Required	Type	Description
\$orderby	No	string	Field name to be used for sorting the result set in ascending order; valid fields are id (property type ID) and <a href="#">_time</a> . To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$select	No	string	Select condition to be applied; valid fields are id (property type ID) and <a href="#">_time</a> .
\$filter	No	string	Filter condition to be applied; valid fields are id (property type ID) and <a href="#">_time</a> .

## Request Example

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/ReferenceProperties/  
core.automobiles:car/wheel_fl
```

Retrieves reference data available for the last one hour for the property set **wheel\_fl** of the thing type **core.automobiles:car**

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/ReferenceProperties/  
core.automobiles:car/wheel_fl?$filter = _time eq 2015-08-01T09:00:00Z
```

Retrieves reference data available at "2015-08-01T09:00:00Z" for the property set **wheel\_fl** of the thing type **core.automobiles:car**

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/ReferenceProperties/  
core.automobiles:car/wheel_fl?$filter = _time gt 2015-08-01T09:00:00Z and _time lt  
2015-08-01T09:10:00Z
```

Retrieves reference data available for a time range "2015-08-01T09:00:00Z" and "2015-08-01T09:10:00Z" for the property set **wheel\_fl** of the thing type **core.automobiles:car**

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
200	Reference data retrieved successfully.

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the structure as shown in the following example:

Request URL `/Things('503B8711B5D0445688E7FF7ACFC2177A')/ReferenceProperties/core.automobiles:car/wheel_fl` retrieves data available for the last one hour as time range is not specified.

```
{
  "value": [{
    "_time": "2015-08-01T09:00:00Z",
    "PressureUnitOfMeasure": "P",
    "RotationUnitOfMeasure": "A",
    "PressureLowerThreshold": 10,
    "RotationLowerThreshold": 15
  }, {
    "_time": "2015-08-01T10:00:00Z",
    "PressureUnitOfMeasure": "PP",
    "RotationUnitOfMeasure": "AA",
    "PressureLowerThreshold": 20,
    "RotationLowerThreshold": 25
  }]
}
```

## 14.3 Read Reference Properties Data for a List of Property Sets

You can use this service to read reference properties data for a list of property sets.

By default, the service retrieves data available for the last one hour. You can use `timerange` property to retrieve data for a specific time range. The time value for the `timerange` property in the request URL must correspond to UTC timestamp.

## Request

**URI:** /Things('<ID>')/<thing type name>/ReferenceProperties/'<property set ID1>,<property type ID1>&<property set ID1>,<property type ID2>&<property set ID2>,<property type ID2>'?timerange=<fromTime>-<toTime>

**HTTP Method:** [GET](#)

**Permissions:** <thing>.r

### Request Header Parameters

Parameter	Required	Values
Accept-Language	No	Language of the reference data. The default language is en.
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header CONTENT-ENCODING with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

### Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thing type name	Yes	string	Name of the thing type
property set ID	Yes	string	Unique identifier of the property type (that belongs to the property set type) defined in a thing type
Property type ID	Yes	string	Name of the property type defined in the property set type

### Request Example

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/ core.automobiles:car/  
ReferenceProperties/'wheel_rr,Pressure&wheel_rr,RotationSpeed&wheel_fl,Pressure&  
wheel_fl,RotationSpeed'?timerange=<fromTime> - <toTime>
```

Reference data is retrieved for the following reference properties:

- **Pressure** and **RotationSpeed** of property set **wheel\_fl**
- **Pressure** and **RotationSpeed** of property set **wheel\_rr**

Both the property set **wheel\_fl** and **wheel\_rr** are assigned to the thing type **core.automobiles:car**.

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
200	Reference data retrieved successfully.

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  "values": {
    "/wheel_rr/Pressure": {
      "com.sap.iot.core.UoM": null,
      "com.sap.iot.core.LowerThreshold": null,
      "com.sap.iot.core.UpperThreshold": null,
      "com.sap.iot.core.LowLowerThreshold": null,
      "com.sap.iot.core.UpperUpperThreshold": null
    },
    "/wheel_rr/RotationSpeed": {
      "com.sap.iot.core.UoM": {
        "id": "RotationUnitOfMeasure",
        "value": [{
          "time": "2016-08-01T09:00:00Z",
          "value": "P"
        }]
      },
      "com.sap.iot.core.LowerThreshold": {
        "id": "PressureLowerThreshold",
        "value": [{
          "time": "2016-08-01T09:00:00Z",
          "value": 10
        }, {
          "time": "2015-08-01T10:00:00Z",
          "value": 15
        }]
      },
      "com.sap.iot.core.UpperThreshold": null,
      "com.sap.iot.core.LowLowerThreshold": null,
      "com.sap.iot.core.UpperUpperThreshold": null
    },
    "/wheel_fl/RotationSpeed": {
      "com.sap.iot.core.UoM": {
        "id": "RotationUnitOfMeasure",
        "value": [{
          "time": "2016-08-01T09:00:00Z",
          "value": "P"
        }, {
          "time": "2015-08-01T10:00:00Z",
          "value": "A"
        }]
      },
      "com.sap.iot.core.LowerThreshold": {
        "id": "RotationLowerThreshold",
```

```

        "value": [{
            "_time": "2016-08-01T09:00:00Z",
            "value": 10
        }]
    },
    "com.sap.iot.core.UpperThreshold": {
        "_id": "RotationUpperThreshold",
        "value": [{
            "_time": "2016-08-01T09:00:00Z",
            "value": 20
        }]
    },
    "com.sap.iot.core.LowLowerThreshold": null,
    "com.sap.iot.core.UpperUpperThreshold": null
},
"/wheel_fl/Pressure": {
    "com.sap.iot.core.UoM": {
        "_id": "PressureUnitOfMeasure",
        "value": [{
            "_time": "2016-08-01T09:00:00Z",
            "value": "P"
        }], {
            "_time": "2015-08-01T10:00:00Z",
            "value": "A"
        }
    ],
    "com.sap.iot.core.LowerThreshold": {
        "_id": "PressureLowerThreshold",
        "value": [{
            "_time": "2016-08-01T09:00:00Z",
            "value": 10
        }], {
            "_time": "2015-08-01T10:00:00Z",
            "value": 15
        }
    ],
    "com.sap.iot.core.UpperThreshold": {
        "_id": "PressureUpperThreshold",
        "value": [{
            "_time": "2016-08-01T09:00:00Z",
            "value": 20
        }], {
            "_time": "2015-08-01T10:00:00Z",
            "value": 25
        }
    ],
    "com.sap.iot.core.LowLowerThreshold": {
        "_id": "PressureLowLowerThreshold",
        "value": [{
            "_time": "2016-08-01T09:00:00Z",
            "value": 5
        }], {
            "_time": "2015-08-01T10:00:00Z",
            "value": 10
        }
    ],
    "com.sap.iot.core.UpperUpperThreshold": {
        "_id": "PressureUpperUpperThreshold",
        "value": [{
            "_time": "2016-08-01T09:00:00Z",
            "value": 30
        }], {
            "_time": "2015-08-01T10:00:00Z",
            "value": 35
        }
    ]
}
}
}

```

```
}
```

## 14.4 Delete Data for Reference Properties

Deletes reference properties data

You can use this service to delete data for reference properties of a specific property set that is assigned to a thing type. The `_time` property is mandatory for deleting reference properties data. If you do not specify `_time`, the system displays an error message.

### Request

**URI:** `/Things('<ID>')/ReferenceProperties/<thing type name>/<property set ID>?<br>$filter=_time eq <time>`

**HTTP Method:** [DELETE](#)

**Permissions:** `<thing>.d`

### Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thing type name	Yes	String	Name of the thing type
property set ID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type
_time	Yes	Timestamp	Time stamp for the properties.

### Query String Parameters

Parameter	Required	Type	Description
\$filter	Yes	String	Filter condition to be applied; the only valid field is <code>_time</code> . The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , and <a href="#">gt</a> . Multiple filters are supported using the <a href="#">AND</a> separator.



## Request Example

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/ReferenceProperties/  
core.automobiles:car/wheel_fl?$filter=_time eq 2009-05-10T09:00:00.000Z
```

Reference data available at the specified time 2009-05-10T09:00:00.000Z is deleted for the property set wheel\_fl of the thing type core.automobiles:car.

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/ReferenceProperties/  
core.automobiles:car/wheel_fl?$filter=_time gt 2016-10-06T09:00:00.000Z and  
_time lt 2016-10-11T09:00:00.000Z
```

Reference data available for the time interval 2016-10-06T09:00:00.000Z and 2016-10-11T09:00:00.000Z is deleted for the property set wheel\_fl of the thing type core.automobiles:car.

## Response

### Response Status and Error Codes

Code	Description
204	Reference data deleted successfully.

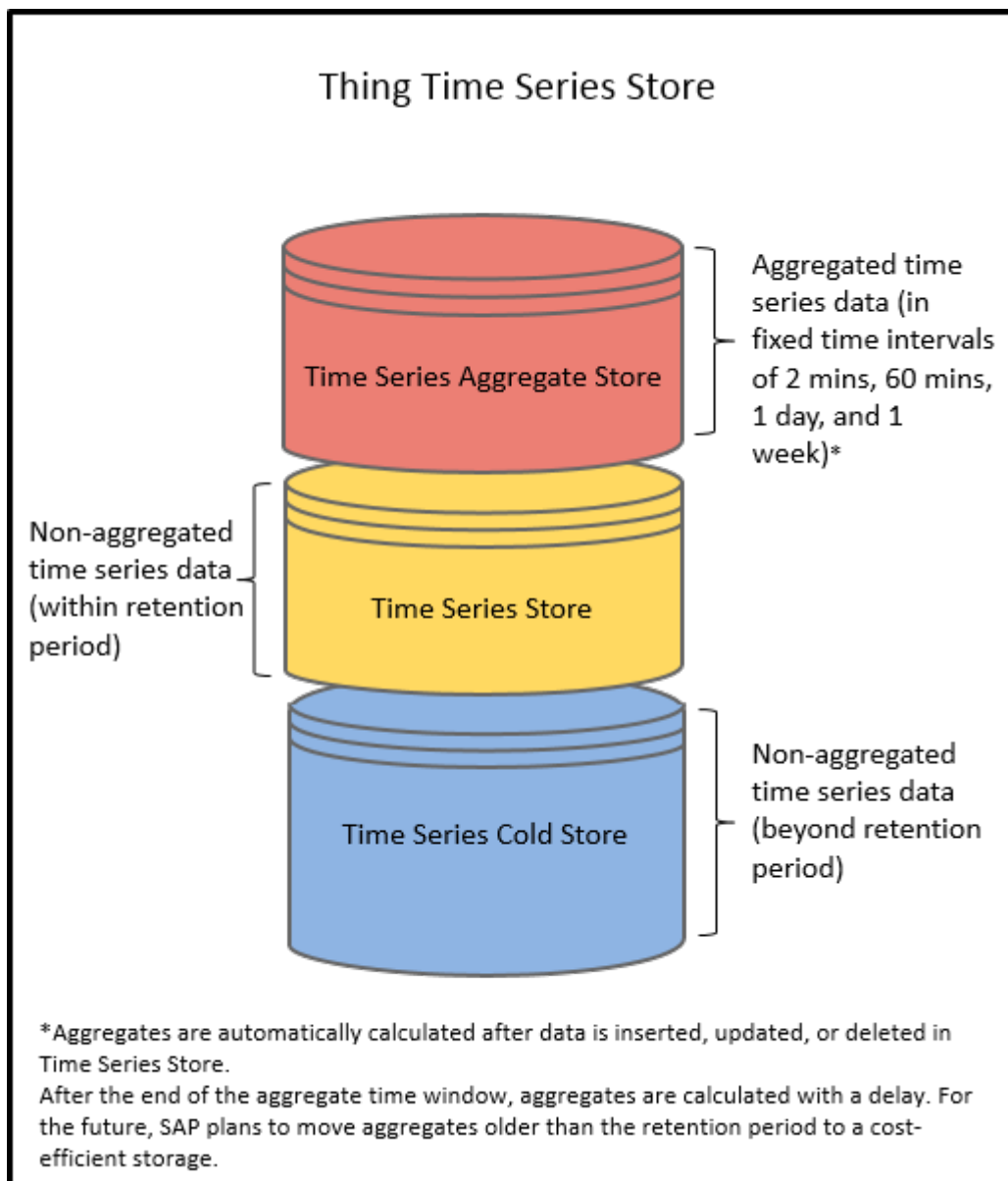
# 15 Thing: Time Series Data

## Thing Time Series Store

The Thing Time Series Store allows you to initialize the time series store for a tenant, manage retention period for a tenant, and manage time series data for Things belonging to a tenant. The time series store uses a tiered storage model containing the following tiers:

- Time Series Store
- Time Series Aggregate Store
- Time Series Cold Store

The following diagram illustrates the Thing Time Series Store:



## Related Information

[Time Series Store \[page 872\]](#)

[Time Series Aggregate Store \[page 890\]](#)

[Time Series Cold Store \[page 910\]](#)

## 15.1 Time Series Store

In an IoT scenario, if the value of a property type changes very frequently, for every second or every hour, you can configure a property set type of data category [TimeSeriesData](#) or [DerivedData](#). For example, you could model the **speed** of a car as [TimeSeriesData](#) and **velocity** or **acceleration** as [DerivedData](#).

The Time Series Store stores non-aggregated time series data for Things of tenant until the retention period configured for the tenant. The non-aggregated time series data beyond the retention period is moved to the Cold Store.

Time series data is always bound to a specific time or time range. You can create time series data for a specific time or retrieve time series data for a specific time range that lies between 1970-01-01T00:00:00Z and current time + 90 days.

The following example describes how the SAP Leonardo IoT services handles, stores, and retrieves the time series data.

- Tenant T1 with retention period configured as 30 days
- Allowed time range for creating time series data is 1970-01-01T00:00:00Z until current time + 90 days (or) 1970-01-01T00:00:00Z and 2018-11-27T00:00:00Z (assuming, current time = 2018-08-30)

Data creation time	Result
2018-11-28T00:00:00Z	Invalid time value as it is beyond the allowed creation time range
2018-10-01T00:00:00Z	Data created in time series store
2018-07-29T00:00:00Z	Data created in cold store as the time lies within 1970-01-01T00:00:00Z and current time - 30 days of retention period

- Allowed time range for retrieving time series data is 1970-01-01T00:00:00Z until current time + 90 days (or) 1970-01-01T00:00:00Z and 2018-11-27T00:00:00Z (assuming, current time = 2018-08-30)

Time range to retrieve data	Result
2018-11-01T00:00:00Z - 2018-12-02T00:00:00Z	Invalid time value as it is beyond the allowed time range
2018-09-01T00:00:00Z - 2018-09-30T00:00:00Z	Data not available in the time series store for the specified time range
2018-11-01T00:00:00Z - 2018-11-10T00:00:00Z	Data not available in the time series store for the specified time range  Inform the user to retrieve the data from cold store

You can create time series data for a thing using the REST-based API [Create Time Series Data or Derived Data for a Thing \[page 877\]](#) or you can use SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (Internet of Things Service) for importing time series data into SAP Leonardo IoT by using the data ingestion services provided by the SAP Leonardo IoT API.

## Base URI

- **Formal description:** `http://<server address>[:<port number>][</path>]`
- **Example for a base URI in a cloud foundry environment:** `https://apiot-mds.cfapps.eu10.hana.ondemand.com/Things`

## Methods

HTTP Method	Action	URI	Scopes
<i>POST</i>	<a href="#">Create Time Series Data Store for Tenant [page 873]</a>	<code>/InitializeTimeseriesStore</code>	<code>&lt;thing&gt;.TenantTSS.c</code>
<i>POST</i>	<a href="#">Create Retention Period for Time Series Data Store [page 874]</a>	<code>/TimeseriesRetentionPeriod</code>	<code>&lt;thing&gt;.Conf.c</code>
<i>GET</i>	<a href="#">Read Retention Period of Time Series Data [page 876]</a>	<code>/TimeseriesRetentionPeriod</code>	<code>&lt;thing&gt;.Conf.r</code>
<i>PUT</i>	<a href="#">Create Time Series Data or Derived Data for a Thing [page 877]</a>	<code>/Things('&lt;thing ID&gt;')/&lt;thing type name&gt;/&lt;property set ID&gt;</code>	<code>&lt;thing&gt;.c</code>
<i>GET</i>	<a href="#">Read Time Series Data or Derived Data for a Thing [page 883]</a>	<code>/Things('&lt;thing ID&gt;')/&lt;thing type name&gt;/&lt;property set ID&gt;</code>	<code>&lt;thing&gt;.r</code>
<i>DELETE</i>	<a href="#">Delete Time Series Data or Derived Data [page 889]</a>	<code>/Things('&lt;thing ID&gt;')/&lt;property set ID&gt;?time-range=&lt;From time&gt;-&lt;To time&gt; (OR) &lt;Thing Application&gt;/URL/Things('&lt;thing ID&gt;')/&lt;property set ID&gt;? time-range=&lt;1H, 1D, 1W, 1M&gt;</code>	<code>&lt;thing&gt;.d</code>

### Note

`<thing>` refers to the thing application name.

## 15.1.1 Create Time Series Data Store for Tenant

Initialize the time series data store for a specific tenant

With this service, you can initialize the time series data store for a specific tenant. You can call this service when you onboard a tenant. After onboarding the tenant and initializing the corresponding time series data store, you can use the store to save the time series data for that specific tenant.

## Request

URI: /InitializeTimeseriesStore

HTTP Method: *POST*

Permissions: <thing>.TenantTSS.c

### Request Parameters

None

### Request Example

```
/InitializeTimeseriesStore
```

## Payload

In the payload of the request, you specify the tenant name for which you want to initialize the time series data store in the <\_tenant> field.

```
{
    "_tenant" : "DEFAULT_TENANT"
}
```

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
201	Time series data store created successfully

## 15.1.2 Create Retention Period for Time Series Data Store

Describes the retention period

With this service, you can define the retention period (in number of days) of the time series data for a specific tenant. Time series data for that specific tenant gets saved in the Time Series Store for the specified retention period. The retention period is calculated based on the business timestamp. For example, time series data is available at the business time value 2016-08-20 and you have defined the retention period as 30 days. The time series data is saved in the Time Series Store until 2016-09-19 (business time value: 2016-08-20 + 30 days).

### Note

If you do not maintain retention period for a specific tenant, a default value of 90 days is saved as retention period.

You can change the retention period for a tenant. However, the changed retention period is applicable only for the time series data ingested after the change in retention period. For the already existing time series data, only the retention period defined at the time the data was written is applicable.

## Request

**URI:** `/TimeseriesRetentionPeriod`

**HTTP Method:** `POST`

**Permissions:** `<thing>.Conf.c`

### Request Parameters

Parameter	Required	Data Type	Description
<code>_retentionPeriodInDays</code>	Yes	Integer	Number of days to retain time series data in Time Series Store: <ul style="list-style-type: none"><li>• Minimum Value: 7 days</li><li>• Maximum Value: 1827 days</li></ul>

### Request Example

`/TimeseriesRetentionPeriod`

## Payload

**Format:** `JSON`

Only media type `JSON` is supported. The `JSON` for this method has the following structure:

```
{
  "_retentionPeriodInDays" : 60
}
```

## Response

**Format:** `JSON`

## Response Status and Error Codes

Code	Description
201	Retention period saved successfully

## 15.1.3 Read Retention Period of Time Series Data

Retrieves retention period of time series data

With this service, you can retrieve the retention period maintained for the time series data for a specific tenant.

### Request

URI: `/TimeseriesRetentionPeriod`

HTTP Method: [GET](#)

Permissions: `<thing>.Conf.r`

### Request Parameters

Parameter	Required	Data Type	Description
<code>_retentionPeriodInDays</code>	Yes	Integer	Number of days to retain time series data in Time Series Store

**i Note**  
If you have not defined the retention period, the service retrieves 90 days as the default retention period.

### Request Example

```
/TimeseriesRetentionPeriod
```

### Response

Format: [JSON](#)

## Response Status and Error Codes

Code	Description
200	Retention time period retrieved successfully



## Payload

**Format:** *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  "_retentionPeriodInDays" : 60
}
```

### 15.1.4 Create Time Series Data or Derived Data for a Thing

Create time series data or derived data for a thing

With this service, you can create data for the property sets of a thing that belongs to a specific thing type. This service supports data creation for the property sets of data categories *TimeSeriesData* and *DerivedData*.

Using this service, you can send the request payload in *JSON* format or as a binary file in *zip* or *gzip* format. The complete request payload must be compressed in a single binary file and you cannot split the request payload in multiple files when using *zip* file format.

#### Time series data ingested via Internet of Things Service

In addition, you can use SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (Internet of Things Service) for importing time series data into SAP Leonardo IoT by using the data ingestion services provided by the SAP Leonardo IoT API.

- When you create a series of time series data for a thing using the data ingestion services, the latest data creation time for the specified thing ID is saved and included in the response payloads of composite APIs.
- When you use the data ingestion pipeline, you can create time series data for a property of data type *GeoJSON*. The *GeoJSON* property type in SAP Leonardo IoT is mapped to a string property type in Internet of Things Service. The value of the *GeoJSON* property type is retrieved in the [Read Time Series Data or Derived Data for a Thing \[page 883\]](#) and [Read Snapshot Data of a Thing \[page 901\]](#) APIs.

The following is an example payload of sensor data sent from Internet of Things Service to SAP Leonardo IoT via data ingestion pipeline:

```
{
  "measures": [{
    "StringGeoValue": "{\"type\":\"Point\",\"coordinates\":
[78.47444,17.37528]}"
  }]
}
```

#### Note

- You can only store latitude, longitude, and geometry type values for a property of data type *GeoJSON*. The only supported geometry type is *Point*.
- In the above example, the property *StringGeoValue* is string data type in Internet of Things Service.

## → Recommendation

You can use this service to create time series data for evaluation purpose. In a real-time scenario or in your production environment, you use the device management system, for example, SAP Cloud Platform Internet of Things for the Cloud Foundry Environment, to ingest the sensor data into the time series store from the physical device.

## Request

**URI:** `/Things('<ID>')/<thing type name>/<property set ID>`

### i Note

- The thing type has to be a fully qualified name that is prefixed with the package name.
- The `_time` property is mandatory for the creation of data of categories `TimeSeriesData` and `DerivedData`. The value for the `_time` property must correspond to UTC timestamp and must lie within a time range from 01.01.1970 to current time + 90 days. Following are the supported formats:
  - `"/Date(<ticks>)/"` `<ticks>` = number of milliseconds since midnight Jan 1, 1970  
For example, `"/Date(1546322403148)/"`
  - `"YYYY-MM-DD HH:MM:SS[.fraction]"`  
For example, `"2015-08-01T09:00:00.000Z"`
- The property name must not contain any special character other than underscore (`_`) and it must not start with a digit.
- You can only create time series data corresponding to a point in time within the retention period specified for the tenant using this service or ingesting via SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (Internet of Things service). When you use this service, if the time stamp is outside retention period, an error occurs. When you ingest data via SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (Internet of Things service), if the time stamp is outside retention period, data is stored in the error log. You can be read the erroneous using ErrorMessages API service - [Read all Error Messages \[page 799\]](#).
- This method is designed to create real-time data. Therefore, we highly recommend not using this method to create huge amounts of historical data. To create data corresponding to time that is not within the retention period specified for the tenant, use [Time Series Cold Store \[page 910\]](#).
- If you have defined quality codes for a specific property but have not assigned values, the system assigns the default value 0 for the quality code property. Quality codes are only supported for properties of data type [Numeric](#), [NumericFlexible](#), or [String](#).
- For property set type of data categories [TimeSeriesData](#) and [DerivedData](#), you cannot create time series data or derived data for a property of type [GeoJSON](#).

**HTTP Method:** [PUT](#)

**Permissions:** `<thing>.c`

## Request Header Parameters

Header	Required	Description
content-Encoding	No	<p>application/zip or application/gzip</p> <div><b>i Note</b> Indicates the content (request payload) to be processed is in application/zip or application/gzip file format.</div>
Content-Type	No	<p>binary/octet-stream</p> <div><b>i Note</b> Indicates the content (request payload) to be processed in this service is in the form of a file.</div>
sap-iot-issync	No	<p>Indicates whether the service saves the latest data creation time for the specified thing ID.</p> <p>Permissible values are true or false</p> <div><b>i Note</b> You can use this header parameter in <b>Thing Detail</b> and <b>Advanced Thing List</b> services.</div>

## Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	<p>You can specify one of the following values:</p> <ul style="list-style-type: none"><li>Unique thing identifier generated by the system while creating a thing.</li><li>Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thingTypeName	Yes	String	Name of a thing type.
propertySetID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

## Request Example

```
/Things('<ID>')/core.automobiles:ABCXSeries/wheel_fl
```

## Payload

Format: *JSON*

Only media type *JSON* is supported. In addition, you can encode the request payload in *JSON* format to *zip* or *gzip* file format using the request header parameter `content-Encoding`. The *JSON* for this method has the structure as specified in the following examples.

### Example 1

Request URL `/Things('C5D30DAAB52C4F01B14C169199BFE0CF')/core.automobiles:ABCXSeries/wheel_fl` to create data for the property set `wheel_fl` of thing `C5D30DAAB52C4F01B14C169199BFE0CF`. In this example, the header parameter `sap-iot-issync` is not specified in the request header. Hence, the latest timestamp value for the thing ID is not saved.

```
{
  "value": [{
    "_time": "2015-08-01T09:00:00Z",
    "Pressure": "36",
    "Temperature": "100"
  }, {
    "_time": "2015-08-02T06:00:03Z",
    "Pressure": "34",
    "Temperature": "120"
  }]
}
```

### Example 2

Request URL `/Things('0E121F2106A94BCBB1D2413D4575386A')/core.automobiles:ABCXSeries/Engine` to create data for the property set **Engine** of thing `0E121F2106A94BCBB1D2413D4575386A`. In this example, the header parameter `sap-iot-issync` is set `true`. Hence, the latest timestamp value `2015-08-02T06:00:03z` is saved and can be retrieved in the composite APIs.

```
{
  "value": [{
    "_time": "2015-08-01T09:00:00Z",
    "pressure": "40",
    "speed": "120"
  }, {
    "_time": "2015-08-02T06:00:03Z",
    "pressure": "28",
    "speed": "108"
  }]
}
```

## Response

Format: *JSON*

## Response Status and Error Codes

Code	Description
200	Data of category <i>TimeSeriesData</i> or <i>DerivedData</i> for the specified thing created successfully

## Related Information

[Thing Detail \[page 1121\]](#)

[Advanced Thing List \[page 1136\]](#)

### 15.1.5 Create Time Series Data or Derived Data Using Alternate Thing ID

With this service, you can create data for the property sets of a thing that belongs to a specific thing type. You must provide the alternate thing ID in the request URL. This service supports data creation for the property sets of data categories *TimeSeriesData* and *DerivedData*.

#### Note

- The thing type has to be a fully qualified name that is prefixed with the package name.
- The `_time` property is mandatory for the creation of data of categories *TimeSeriesData* and *DerivedData*. The value for the `_time` property must correspond to UTC timestamp and must be less than or equal to `current time + 90 days`.

## Request

**URI:** `/ThingsByAlternateId('<alternate thing ID>')/<thing type name>/<property set ID>`

**HTTP Method:** *PUT*

**Permissions:** `<thing>.c`

## Request Header Parameters

Header	Required	Description
sap-iot-issync	No	Indicates whether the service saves the latest data creation time for the specified thing ID.  Permissible values are <b>true</b> or <b>false</b>

**Note**  
You can use this header parameter in **Thing Detail** and **Advanced Thing List** services.

## Request Parameters

Parameter	Required	Data Type	Description
alternate thing ID	Yes	String	Alternate thing identifier unique within a specific tenant.
thing type name	Yes	String	Name of a thing type.
property set ID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type.

## Request Example

```
/ThingsByAlternateId('ABCXSeries001-alternateID')/core.automobiles:ABCXSeries/wheel_fl
```

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  "value": [{
    "_time": "2015-08-01T09:00:00Z",
    "Pressure": "36",
    "Temperature": "100"
  }]
}
```

## Response

### Response Status and Error Codes

Code	Description
200	Data of category <i>TimeSeriesData</i> or <i>DerivedData</i> for the specified thing created successfully.

### Related Information

[Create Time Series Data or Derived Data for a Thing \[page 877\]](#)

## 15.1.6 Read Time Series Data or Derived Data for a Thing

Retrieves property set data for the specified thing

With this service, you can retrieve the values of property sets of a thing that belongs to a specific thing type. You can request the values for a specific time range or use the filter condition on `_time`. The service retrieves the values of property sets of data categories *TimeSeriesData* or *DerivedData*.

#### Note

Retrieving number of objects more than the recommended limits with only one collective GET request can lead to high system load and a significant increase of response times.

You cannot specify time range and use `$filter` on `_time` in the request URL to retrieve time series data or derived data. For example, the request URL `/Things('<ID>')/<thing_type name>/<property set ID>?timerange=2017-07-02T00:00:00Z-2017-07-02T10:00:00Z & $filter=_time ge 2017-07-02T00:00:00Z and _time le 2017-07-02T10:00:00Z` is invalid.

The following table illustrates the various combinations of time range and `$filter` on `_time` in the request URL:

Request URL with	Example	Result
Valid time range and <code>\$filter</code> on <code>_time</code>	<code>/Things('&lt;ID&gt;') / &lt;thing_type name&gt; / &lt;property set ID&gt; ? timerange=2017-07-02T00:00:00Z-2017-07-02T10:00:00Z &amp; \$filter=_time ge 2017-07-02T00:00:00Z and _time le 2017-07-02T10:00:00Z</code>	Invalid Response status: 400 Bad Request

Request URL with	Example	Result
Valid time range but no \$filter on _time	<code>/Things('&lt;ID&gt;') / &lt;thing_type name &gt; / &lt;property set ID&gt;? timerange=2017-07-02T00:00:00Z-2017-07-02T10:00:00Z</code>	Data is fetched for the specified time range 2017-07-02T00:00:00Z to 2017-07-02T10:00:00Z
No valid time range but \$filter on _time	<code>/Things('&lt;ID&gt;') / &lt;thing_type name&gt; / &lt;property set ID&gt;? \$filter=_time ge 2017-07-02T00:00:00Z and _time le 2017-07-02T10:00:00Z on valid</code>	Data is fetched for the time range 2017-07-02T00:00:00Z to 2017-07-02T10:00:00Z specified in \$filter
No valid time range and no \$filter on _time	<code>/Things('&lt;ID&gt;') / &lt;thing_type name&gt; / &lt;property set ID&gt;</code>	Data is fetched for the last one hour

### Note

- You cannot use the filter operator *ne* for filter condition on `_time`. For example, `/Things('<ID>') /<thing_type name> /<property set ID>?$filter=_time ne 2017-07-02T00:00:00Z` is not a valid request.
- You cannot use parenthesis to group filter condition on `_time`. For example, `/Things('<ID>') /<thing_type name> /<property set ID>?$filter=_ (time ne 2017-07-02T00:00:00Z)` is not a valid request.
- You cannot group the filter conditions on `_time` and any other property. For example, `/Things('<ID>') /<thing_type name> /<property set ID>?$filter=_property1 gt 10 or time eq 2017-07-02T00:00:00Z` is not a valid request.
- You cannot group filter conditions on `_time` using the filter operator *eq* along with the separator *or*. For example, `/Things('<ID>') /<thing_type name> /<property set ID>?$filter=_time eq 2017-01-01T00:00:00Z or _time eq 2017-01-05T00:00:00Z` is not a valid request.

## Time series data ingested via Internet of Things Service

You can use SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (Internet of Things Service) for importing time series data into SAP Leonardo IoT by using the data ingestion services provided by the SAP Leonardo IoT API. When sensor data from the physical device is imported into SAP Leonardo IoT, Internet of Things Service creates the necessary payload that contains information such as `capabilityAlternateId`, `sensorAlternateId`, and `timestamp` along with the message received from the device.

However, in certain scenarios, collection of measured values can be sent from the device along with the actual timestamp at which the data was generated in the JSON message payload.

```
{
  "capabilityAlternateId": 19,
  "sensorAlternateId": "124223",
  "timestamp": "144740714257",      // automatically inserted by SCP IoT Service
  "measures": [{
```



```

    "roll": "27.63",
    "pitch": "28.63",
    "yaw": "29.63",
    "_time": 1447407100060
  }
}

```

To import the actual time at which the sensor data was generated, you must define the `_time` property for the capability in Internet of Things Service. However, it is not mandatory to define this property and import the actual time. In addition, you must not map the `_time` property to a corresponding property in **Thing Modeler app**, unlike other properties. The data ingestion services provided by the SAP Leonardo IoT API only accepts [Long](#) values or values that are parsable to [Long](#) for the `_time` property. The [Long](#) value contained in the `_time` property is considered as [Epoch Milliseconds](#) (number of milliseconds elapsed since 1970-01-01 00:00:00 GMT). The data ingestion service rejects a record with `_time` value that cannot be parsed as Long.

When you retrieve data for the specified thing, the service retrieves data as follows:

- If `_time` property with value is available along with the measured values sent from the device in the JSON message payload, the service uses `_time` as the business timestamp.
- If the `_time` property is not available along with the measured values sent from the device in the JSON message payload, the service uses `timestamp` as the business timestamp.
- If both `_time` and `timestamp` are not available in the JSON message payload received from the device, the service uses the current time at which the record is being processed as the business timestamp.
- If `_time` contains an invalid value, the service does not process the JSON message payload received from the device.

## Request

URI: `/Things('<ID>')/<thing type name>/<property set ID>`

### Note

- The thing type has to be a fully qualified name that is prefixed with the package.
- The service only supports the time range value lying between 01.01.1970 and current time + 90 days.
- If you do not specify a time range, the data that is available for the last one hour is retrieved. If the data is not available for the last one hour, no data is retrieved.
- The service only retrieves data corresponding to a time range within the retention period specified for the tenant. An error occurs if the time range is not within the retention period.

HTTP Method: [GET](#)

Permissions: `<thing>.r`

## Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header <code>CONTENT-ENCODING</code> with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response

## Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thingTypeName	Yes	String	Name of the thing type.
propertySetID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

## Query String Parameters

Parameter	Required	Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; only valid field is <code>_time</code> . <div><b>i Note</b> Data types <a href="#">JSON</a> and <a href="#">ByteArray</a> are not supported.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</div>

Parameter	Required	Type	Description
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are <code>_time</code> and all the property types.
\$filter	No	String	Filter condition to be applied; all the fields in the result set are valid.

**Note**

Data types *JSON* and *ByteArray* are not supported.

Multiple filters are supported using the *and* separator.

## Request Example

```
/Things('<ID>')/core.automobiles:ABCXSeries/wheel_fl?timerange=10M
```

Retrieves property set data of the property set `wheel_fl` for the thing ID `core.automobiles:ABCXSeries` corresponding to a time range of 10 months.

```
/Things('<ID>')/core.automobiles:ABCXSeries/wheel_fl?timerange=10M&$select=_time,Pressure
```

Retrieves property set data of the property set `wheel_fl` for the thing ID `core.automobiles:ABCXSeries` corresponding to a time range of 10 months. However, with *\$select* query parameter, only the fields `_time` and `Pressure` are displayed for the retrieved thing.

```
/Things('<ID>')/core.automobiles:ABCXSeries/wheel_fl?timerange=2015-08-01T09:00:00Z-2015-10-01T09:00:00Z
```

Retrieves property set data of the property set `wheel_fl` for the thing ID `core.automobiles:ABCXSeries` corresponding to a time range from August 1, 2015 to October 1, 2015.

## Response

### Response Properties

Property	Type	Description
Value	Array of thing data objects	An array of data objects (for example, time series data) for a thing or property set type.

### Response Status and Error Codes

Format: *JSON*

Code	Description
200	Data for one or more things retrieved successfully

## Payload

**Format:** *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

### Example 1

Retrieves property set data of the property set `wheel_fl` for the thing ID `core.automobiles:ABCXSeries` corresponding to a time range from August 1, 2015 to August 2, 2015.

```
{
  "value": [{
    "_time": "2015-08-01T09:00:00Z",
    "Pressure": "36",
    "Temperature": "100"
  }, {
    "_time": "2015-08-02T06:00:03Z",
    "Pressure": "34",
    "Temperature": "120"
  }]
}
```

### Example 2

Retrieves property set data of the property set `StringGeoValue` for the thing ID `core.automobiles:ABCXSeries`. The property `StringGeoValue` is of data type `GeoJSON`.

```
{
  "value": [{
    "_time": "2019-02-11T13:13:16.967Z",
    "StringGeoValue": {
      "position": {
        "coordinate": [
          78.47444,
          17.37528
        ],
        "dimension": 2
      },
      "geometryType": "Point"
    }
  }]
}
```

## Related Information

[Time Series Store \[page 872\]](#)  
[Sensors and Things \[page 748\]](#)

## 15.1.7 Delete Time Series Data or Derived Data

Delete time series data or derived data for a thing

With this service, you can delete the time series data for the specified thing and property set type corresponding to the specified time range.

### Request

**URI:** /Things('<ID>')/<thing type name>/<property set ID>?timerange=<From time>-<To time> (OR) /Things('<ID>')/<thing type name>/<property set ID>? timerange=<1H, 1D, 1W, 1M>

#### Note

- You must specify a time range to be able to delete the time series data.
- The service only deletes data corresponding to a time range within the specified retention period of the tenant.
- After deleting time series data, the system recalculates the aggregates.

**HTTP Method:** [DELETE](#)

**Permissions:** <thing>.d

### Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
propertySetID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

### Request Example

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/core.automobiles:ABCXSeries/car?timerange=2015-08-01T06:00:00Z-2015-08-02T09:00:00Z
```

### Response

**Format:** [JSON](#)

## Response Status and Error Codes

Code	Description
204	Data deleted successfully

## 15.2 Time Series Aggregate Store

The Time Series Aggregate Store stores the pre-calculated aggregates for the time series data of Things. The following are the pre-calculated aggregates available for use by the supported thing model APIs:

- Minimum value of a property type with the corresponding time value within a specific time duration
- Maximum value of a property type with the corresponding time value within a specific time duration
- First occurring value of a property type with the corresponding time value within a specific time duration
- Last occurring value of a property type with the corresponding time value within a specific time duration

### Time Intervals Supported for Aggregation

This store contains pre-calculated aggregates based on the following time intervals:

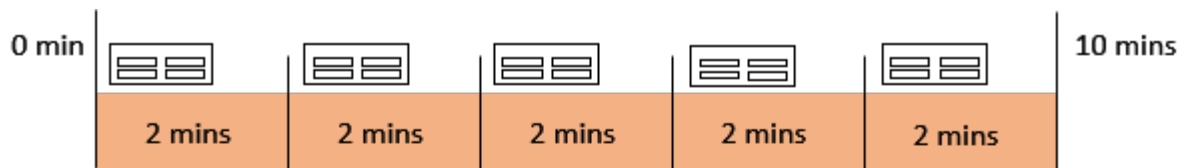
- **2 mins:** Aggregates are calculated based on measurement points available for the time interval between each consecutive even minutes. For example, all measurement points with business time `_time >= 2016-09-01T15:02:00Z` and `_time < 2016-09-01T15:04:00Z` are used to compute the aggregate with time as `2016-09-01T15:02:00Z` and duration as 120000 millisecond.
- **1 hour:** Aggregates are calculated based on measurement points available for the time interval between each consecutive hours. For example, all measurement points with business time `_time >= 2016-09-01T15:00:00Z` and `_time < 2016-09-01T16:00:00Z` are used to compute the aggregate with time as `2016-09-01T15:00:00Z` and duration as 3600000 millisecond.
- **1 day:** Aggregates are calculated based on measurement points available for the time interval between each consecutive days. For example, all measurement points with business time `_time >= 2016-09-01T00:00:00Z` and `_time < 2016-09-02T00:00:00Z` are used to compute the aggregate with time as `2016-09-01T00:00:00Z` and duration as 86400000 millisecond.
- **1 week:** Aggregates are calculated based on measurement points available for the time interval between each consecutive weeks. For example, all measurement points with business time `_time >= 2016-09-01T00:00:00Z` and `_time < 2016-09-08T00:00:00Z` are used to compute the aggregate with time as `2016-09-01T00:00:00Z` and duration as 604800000 millisecond.

#### i Note

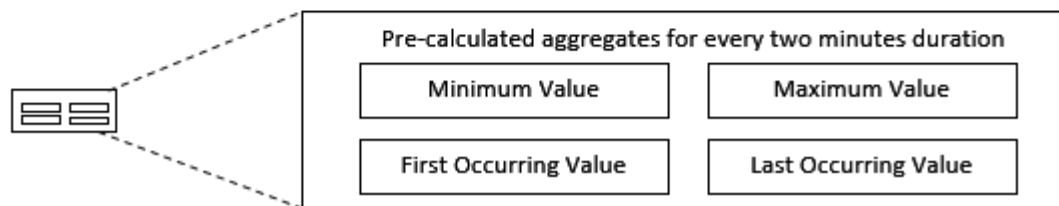
The start day for the weekly aggregate is calculated by considering 1970-01-01T00:00:00Z as base and adding seven days to arrive at the start of the next corresponding weeks.

The following diagram illustrates the time intervals for calculating aggregates in the Time Series Aggregate Store:

#### 2 mins time interval



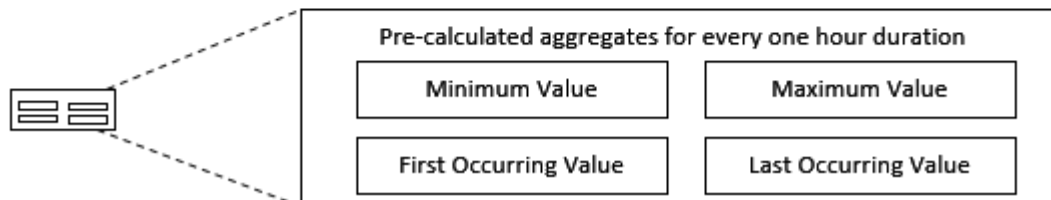
Unique minimum value, maximum value, first occurring value, and last occurring value along with time values are pre-calculated and stored for every consecutive even minutes



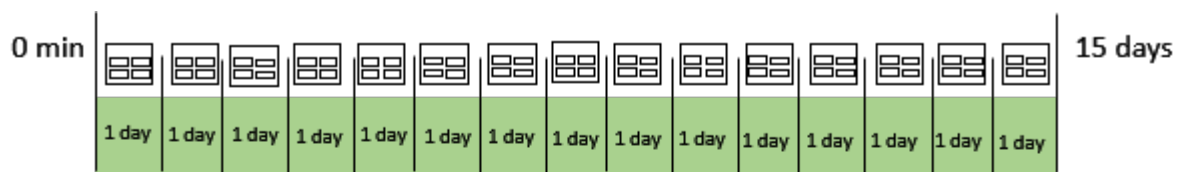
1 hour time interval



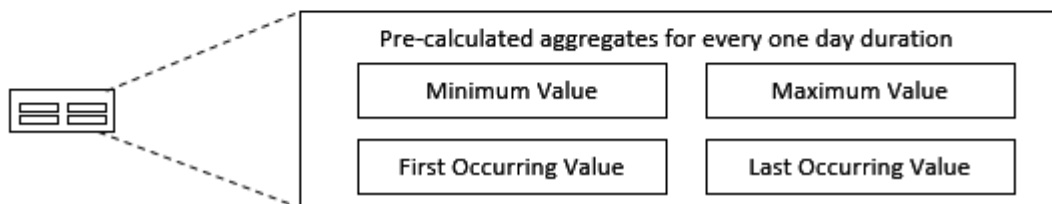
Unique minimum value, maximum value, first occurring value, and last occurring value along with time values are pre-calculated and stored for every consecutive hour



1 day time interval



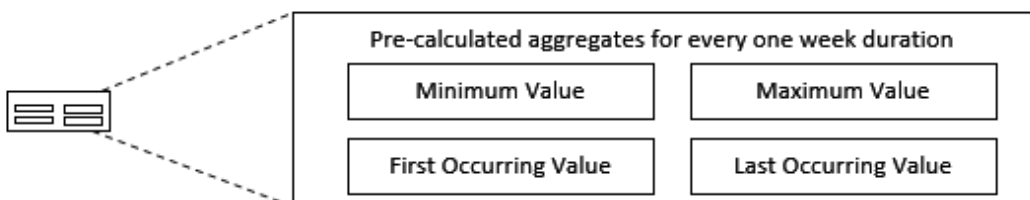
Unique minimum value, maximum value, first occurring value, and last occurring value are pre-calculated and stored for every consecutive day



1 week time interval



Unique minimum value, maximum value, first occurring value, and last occurring value are pre-calculated and stored for every consecutive week



## Time Bucket

The time bucket is the time interval that is calculated as follows:

Time Bucket = (toTime - fromTime) / number of divisions

In addition, the time bucket value is rounded down to multiples of time intervals (durations) supported for aggregate calculation in Time Series Aggregate Store. Similarly, based on the time bucket value calculated, the fromTime is rounded down. Hence, calculation of time bucket and rounding off fromTime may lead to retrieval of additional measuring points than requested.

### Example 1

fromTime: 2016-09-01T15:02:00Z

toTime: 2016-09-02T15:02:00Z

Number of divisions requested: 25



Time Bucket = (2016-09-02T15:02:00Z- 2016-09-01T15:02:00Z) / 25 = 57.6 minutes, which is rounded down to the multiple of nearest duration (2 minutes) supported for aggregate calculation, that is 56 minutes. Note that the time bucket of 56 minutes need not start at 2016-09-01T15:02:00Z but at a multiple of 56 minutes since 1970-01-01T00:00:00Z, that is at 2016-09-01:T14:56:00Z.

### Example 2

fromTime: 2016-08-01T15:02:00Z

toTime: 2016-08-03T16:08:00Z

Number of divisions requested: 10

Time Bucket = (2016-08-03T16:08:00Z- 2016-08-01T15:02:00Z) / 10 = 4 hours 9 minutes 36 seconds, which is rounded down to the multiple of nearest duration (1 hour) supported for aggregate calculation, that is 4 hours. Note that the time bucket of 4 hours need not start at 2016-08-01T15:02:00Z but at a multiple of 1 hour since 1970-01-01T00:00:00Z, that is at 2016-08-01T15:00:00Z

### Example 3

fromTime: 2016-08-15T15:02:00Z

toTime: 2016-09-01T15:02:00Z

Number of divisions requested: 10

Time Bucket = (2016-09-01T15:02:00Z- 2016-08-15T15:02:00Z) / 10 = 1.8 days, which is rounded down to the multiple of nearest duration (1 day) supported for aggregate calculation, that is 1 day. Note that the time bucket of 1 day need not start at 2016-08-15T15:02:00Z but at a multiple of 1 day since 1970-01-01T00:00:00Z, that is at 2016-08-15T00:00:00Z.

### Example 4

fromTime: 2016-08-10T15:02:00Z

toTime: 2016-09-23T15:02:00Z

Number of divisions requested: 3

Time Bucket = (2016-09-01T15:02:00Z- 2016-08-15T15:02:00Z) / 3 = 14.6 days, which is rounded down to the multiple of nearest duration (1 week) supported for aggregate calculation, that is 2 weeks. Note that the time bucket of 2 weeks need not start at 2016-08-10T15:02:00Z but at a multiple of 1 week since 1970-01-01T00:00:00Z, that is at 2016-08-03T00:00:00Z

### Base URI

- **Formal description:** `http://<server address>[:<port number>][[/path]]`
- **Example for a base URI in a cloud foundry environment:** `https://apiiot-mds.cfapps.eu10.hana.ondemand.com/Things`

## Methods

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read Time Series Data for a Property Type of a Thing Applying M4 Algorithm [page 894]</a>	/Things('<thing ID>')/<thing type name>/<property set ID>?timerange=<fromTime>-<toTime>&\$apply=sap.lineChart(divisions=<number of divisions>, property='<propertyID>')	<thing>.r
GET	<a href="#">Read Time Series Data for Multiple Property Types of a Thing Applying M4 Algorithm [page 897]</a>	/Things('<thing ID>')/<thing type name>/M4/'<property set ID1>,<property type ID1>&<property set ID2>&<property set ID2>,<property type ID2>'?timerange=<fromTime>-<toTime>&\$apply=sap.lineChart(divisions=<number of divisions>, property='<propertyID>')	<thing>.r
POST	<a href="#">Recalculate M4 Aggregates for Time Series Data [page 900]</a>	/Things('<thing ID>')/<thing type name>/<property set ID>/RecalculateAggregate?timerange=<fromTime>-<toTime>	<thing>.c
GET	<a href="#">Read Snapshot Data of a Thing [page 901]</a>	/Snap-shot(thingId='<thingID>',fromTime='<timestamp>',dataCategory='<dataCategory>')	<thing>.r
GET	<a href="#">Read Snapshot Data Within a Time Range for a Thing [page 906]</a>	/v2/Snap-shot(thingId='<thingID>',fromTime='<timestamp>',toTime='<timestamp>',dataCategory='<dataCategory>')	<thing>.r

### 15.2.1 Read Time Series Data for a Property Type of a Thing Applying M4 Algorithm

Retrieves time series data for a specified property type

With this service, you can read the time series data for a specific property type of a thing by applying the M4 algorithm. This algorithm is used to aggregate the time series data for n number of divisions. Divisions correspond to the number of intervals into which the given time range needs to be divided. For each interval, the service returns unique minimum value, maximum value, first occurring value, and last occurring value, if they exist. Hence, there would be a maximum of four values per division returned. This service only supports property set types of data categories *TimeSeriesData* and *DerivedData*.

### Note

- This service only supports property types of type [Numeric](#), [NumericFlexible](#), and [Boolean](#).
- If, for a specific division, all the four values (combination of timestamp and property values) are identical, only one value is returned.
- If, for a specific division, the maximum and minimum values are equal while the first and last values are equal, only two values are returned.
- The last record in the specified time range is always retrieved.

The service retrieves data from the pre-calculated aggregates available for every 2 minutes, 60 minutes, 1 day, and 1 week.

## Request

**URI:** /Things('<ID>')/<thing type name>/<property set ID>?timerange=<fromTime>-<toTime>&\$apply=sap.lineChart(divisions=<number of divisions>,property='<propertyID>')

**HTTP Method:** [GET](#)

**Permissions:** <thing>.r

### Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header CONTENT-ENCODING with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

### Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thingTypeName	Yes	String	Name of the thing type.

Parameter	Required	Data Type	Description
propertySetID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type
timerange	Yes	Timestamp	You can provide a from time and to time for which the aggregates are calculated for a property type. Alternatively, you can provide the duration of time in days, months, or years for calculating the aggregate.

## Request Example

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/core.automobiles:ABCXSeries/wheel_fl?
timerange=2015-08-01T06:00:00Z-2015-08-02T09:00:00Z&
$apply=sap.lineChart(divisions=400,property='Speed')
```

Retrieves property set data of the property set wheel\_fl for the thing ID 503B8711B5D0445688E7FF7ACFC2177A corresponding to a time range between 2015-08-01T06:00:00Z and 2015-08-02T09:00:00Z.

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/core.automobiles:ABCXSeries/wheel_fl?
timerange=2D&$apply=sap.lineChart(divisions=400,property='Speed')
```

Retrieves property set data of the property set wheel\_fl for the thing ID 503B8711B5D0445688E7FF7ACFC2177A corresponding to a time duration of 2 days.

## Response

Format: [JSON](#)

## Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [JSON](#)

Only media type [JSON](#) is supported. The [JSON](#) for this method has the following structure:

```
{
  "value": [{
    "_time": "2015-08-01T09:00:00.000Z",
    "WheelBase": "11"
  }, {
    "_time": "2015-08-02T06:00:03.000Z",
    "WheelBase": "10"
  }]
}
```

## Related Information

[Time Series Aggregate Store \[page 890\]](#)

### 15.2.2 Read Time Series Data for Multiple Property Types of a Thing Applying M4 Algorithm

Retrieves time series data for multiple property types

With this service, you can read the time series data for multiple property types of the specified thing by applying the M4 algorithm. This algorithm is used to aggregate the time series data for n number of divisions. Divisions correspond to the number of intervals into which the given time range needs to be divided. For each interval, the service returns unique minimum value, maximum value, first occurring value, and last occurring value, if they exist. Hence, there would be a maximum of four values per division returned. This service only supports property set types of data categories [TimeSeriesData](#) and [DerivedData](#).

#### Note

- This service only supports property types of type [Numeric](#), [NumericFlexible](#), and [Boolean](#).
- If, for a specific division, all the four values (combination of time value and property values) are identical, only one value is returned.
- If, for a specific division, the maximum and minimum values are equal while the first and last values are equal, only two values are returned.
- The service retrieves data from a pre-calculated aggregates available for every 2 minutes, 60 minutes, 1 day, and 1 week.

## Request

**URI:** `/Things('<ID>')/<thing type name>/M4/'<property set ID1>,<property type ID1>&<property set ID1>,<property type ID2>&<property set ID2>,<property type ID2>'?timerange=<fromTime>-<toTime>&$apply=sap.lineChart(divisions=<number of divisions>)`

**HTTP Method:** [GET](#)

**Permissions:** `<thing>.r`

## Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header CONTENT-ENCODING with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thingTypeName	Yes	String	Name of the thing type.
propertySetID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type
timerange	Yes	Timestamp	You can provide a from time and to time for which the aggregates are calculated for a property type. Alternatively, you can provide the duration of time in days, months, or years for calculating the aggregate.

## Request Example

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/core.automobiles:ABCXSeries/  
M4/'wheel_fl,pressure&wheel_fl,temperature&wheel_rr,pressure&wheel_rr,temperature'?  
timerange=2015-08-01T06:00:00Z-2015-08-02T09:00:00Z&  
$apply=sap.lineChart(divisions=400)
```

Retrieves time series data for multiple property types of the specified thing by applying the M4 algorithm. The time range provided is between the from time and to time.

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/core.automobiles:ABCXSeries/  
M4/'wheel_fl,pressure&wheel_fl,temperature&wheel_rr,pressure&wheel_rr,temperature'?  
timerange=2M&$apply=sap.lineChart(divisions=400)
```

Retrieves time series data for multiple property types of the specified thing by applying the M4 algorithm. The time range provided is for a duration of two months.

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  "value": {
    "/engine/capacity": [{
      "_time": "2015-08-01T09:00:00.000Z",
      "value": "11"
    }, {
      "_time": "2015-08-01T09:00:00.000Z",
      "value": "13"
    }],
    "/car/WheelBase": [{
      "_time": "2015-08-01T09:00:00.000Z",
      "value": "23"
    }, {
      "_time": "2015-09-01T09:00:00.000Z",
      "value": "50"
    }],
    "/wheel_rr/pressure": {
      "_time": "2015-08-01T09:00:00.000Z",
      "value": "15"
    },
    "/wheel_fl/pressure": {
      "_time": "2015-08-01T09:00:00.000Z",
      "value": "21"
    },
    "/wheel_fl /temperature": {
      "_time": "2015-08-01T09:00:00.000Z",
      "value": "33"
    },
    "/drive_fl/RotationSpeed": {
      "_time": "2015-08-01T09:00:00.000Z",
      "value": "19"
    },
    "/drive_fl/PowerConsumption": {
      "_time": "2015-08-01T09:00:00.000Z",
      "value": "13"
    }
  }
}
```

## Related Information

[Time Series Aggregate Store \[page 890\]](#)

### 15.2.3 Recalculate M4 Aggregates for Time Series Data

Recalculate all M4 aggregates for a specific time range

With this service, you can recalculate all M4 aggregates for a specific time range. The service only supports the following:

- Data categories [TimeSeriesData](#) and [DerivedData](#)
- Time range less than or equal to one day

#### Note

You can use this service to re-trigger the aggregate calculation if there are errors in the existing aggregates, if there is a delay in the automatically triggered aggregate calculation process, or when you delete the time series data.

## Request

**URI:** /Things('<ID>')/ <thing type name>/<property set ID>/RecalculateAggregate?  
timerange=<fromTime>-<toTime>

**HTTP Method:** [POST](#)

**Permissions:** <thing>.c

## Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header CONTENT-ENCODING with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.



## Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
thingTypeName	Yes	String	Name of the thing type
propertySetID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type
fromTime	Yes	timeStamp	Corresponds to the time value from which the time series data is retrieved for a non-null value of a property
toTime	Yes	timeStamp	Corresponds to the time value until which the time series data is retrieved for a non-null value of a property

## Request Example

```
/Things('503B8711B5D0445688E7FF7ACFC2177A')/core.automobiles:ABCXSeries/wheel_fl/RecalculateAggregate?timerange=2016-10-07T00:00:00.000Z-2016-10-07T00:23:00.000Z
```

## Response

Format: *JSON*

## Response Status and Error Codes

Code	Description
200	Recalculation of aggregates successfully completed

## 15.2.4 Read Snapshot Data of a Thing

Retrieves snapshot data of a thing

With this service, you can retrieve the latest value of properties for the specified thing ID. The service uses *fromTime* to define the time range to retrieve the latest non-null value. If you specify a valid time value for *fromTime*, the service retrieves the latest non-null value for a time range between the current time and the specified *fromTime*. If you do not specify a value or specify an empty value for *fromTime*, the service scans through the data available from the current time until it finds a non-null value. If the data is not available, the service returns null. The service does not support data category *ReferencePropertyData*.

### Note

- The service retrieves the snapshot data for properties that support multiple languages in [Locale](#) specific language. If data is not available, the service returns null for the property.
- If you have configured a property of type `GeoJSON`, and the time series data for this property was created using the data ingestion pipeline, this service includes the property values in the response payload.

## Request

**URI:** `/Snapshot(thingId='<ID>', fromTime='<timestamp>', dataCategory='<dataCategory>')`

### Note

- Sequence of the parameters within the URI must not be changed. For example, `/Snapshot(dataCategory='<dataCategory>', thingId='<ID>', fromTime='<timestamp>')` is an invalid request.
- The thing ID is mandatory to retrieve the snapshot details.
- With a valid value for [fromTime](#), the URI is `/Snapshot(thingId='<ID>', fromTime='<timestamp>', dataCategory='')`
- With an empty value for [fromTime](#), the URI is `/Snapshot(thingId='<ID>', fromTime='', dataCategory='')`

You can retrieve latest value only for specific properties using the query parameter [\\$select](#). The URI is `/Snapshot(thingID='<ID>', fromTime='<timestamp>', dataCategory='<dataCategory>')?$select=<property set ID>/<property ID>`

**HTTP Method:** [GET](#)

**Permissions:** `<thing>.r`

## Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip.deflate</a> for this parameter, the system returns the compressed response with an additional response header <code>CONTENT-ENCODING</code> with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	<p>You can specify one of the following values:</p> <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
fromTime	Yes	timestamp	<p>Time from which the snapshot details are retrieved for a non-null value of a property. The value for <code>fromTime</code> must correspond to UTC timestamp.</p> <div><b>i Note</b> If you do not specify a value or specify an empty value for this parameter, the service scans through the data available from the current time until it finds a non-null value. If the data is not available, the service returns null.</div>
dataCategory	Yes	enum	<p>Filters the snapshot details based on data category enumeration values such as <a href="#">MasterData</a>, <a href="#">DerivedData</a>, <a href="#">TimeSeriesData</a>, and <a href="#">Parameters</a>.</p>
propertySetID	No	string	<p>Property set ID</p> <div><b>i Note</b> In case of thing type created using the REST service, use the property type name defined for the thing type.</div>
propertyID	No	string	<p>Name of the property</p> <div><b>i Note</b> In case of thing type created using the REST service, use the property type name defined for the property set type.</div>

## Request Example

Read snapshot of time series data for a thing

/

```
Snapshot(thingId='503B8711B5D0445688E7FF7ACFC2177A',fromTime='2015-01-01T06:00:00Z',dataCategory='TimeSeriesData')
```

Read snapshot of time series data for specific properties of a thing

```
/Snapshot(thingId='503B8711B5D0445688E7FF7ACFC2177A',fromTime='',dataCategory='')?$select=car/BodyStyle,wheel_rr/Temperature
```

## Response

### Response Properties

Property	Type	Description
<code>_id</code>	string	Internal identifier for a thing
<code>_thingType</code>	Array of thing type objects	Array of thing types

Format: [JSON](#)

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [JSON](#)

Only media type [JSON](#) is supported. The [JSON](#) for this method has the following structure:

### Read snapshot data for a thing

```
{
  "_id": "503B8711B5D0445688E7FF7ACFC2177A",
  "_thingType": [
    "core.automobiles:ABCXSeries"
  ],
  "value": [{
    "core.automobiles:ABCXSeries": [{
      "/car": [{
        "BodyStyle": {
          "_value": "SUV",
          "_qualityCode": 0,
          "_time": "2015-08-02T06:00:03Z",
          "_unitOfMeasure": ""
        }
      }, {
        "Class": {
          "_value": "1st",
          "_qualityCode": 0,
          "_time": "2015-08-02T06:00:03Z",
          "_unitOfMeasure": ""
        }
      }, {
        "WheelBase": {
          "_value": "15",
          "_qualityCode": 9,
          "_time": "2015-08-02T06:00:03Z",
          "_unitOfMeasure": ""
        }
      }, {
        "RegistrationNumber": {
          "_value": "001",
```

```

        "_qualityCode": 0,
        "_time": "2015-08-02T06:00:03Z",
        "_unitOfMeasure": ""
    }, {
        "Colour": {
            "_value": "Black",
            "_qualityCode": 0,
            "_time": "2015-08-02T06:00:03Z",
            "_unitOfMeasure": ""
        }
    }],
    "_propertySetType": {
        "name": "core.automobiles:Car",
        "description": "Sensory parameters for car"
    }
}, {
    "/wheel_rr": [{
        "Pressure": {
            "_value": null,
            "_qualityCode": 0,
            "_time": null,
            "_unitOfMeasure": "PA"
        }
    }, {
        "Temperature": {
            "_value": null,
            "_qualityCode": 0,
            "_time": null,
            "_unitOfMeasure": ""
        }
    }],
    "_propertySetType": {
        "name": "core.automobiles:Wheel",
        "description": "Sensory parameters for wheel"
    }
}
]]
}

```

## Read snapshot data for specific properties of a thing

```

{
  "_id": "503B8711B5D0445688E7FF7ACFC2177A",
  "_thingType": [
    "core.automobiles:ABCXSeries"
  ],
  "value": [{
    "core.automobiles:ABCXSeries": [{
      "/car": [{
        "BodyStyle": {
          "_value": "SUV",
          "_qualityCode": 0,
          "_time": "2015-08-02T06:00:03Z",
          "_unitOfMeasure": ""
        }
      }],
      "_propertySetType": {
        "name": "core.automobiles:Car",
        "description": "Sensory parameters for car"
      }
    }
  ], {
    "/wheel_rr": [{
      "Temperature": {
        "_value": null,
        "_qualityCode": 0,
        "_time": null,

```

```

        "_unitOfMeasure": ""
    },
    ],
    "_propertySetType": {
        "name": "core.automobiles:Wheel",
        "description": "Sensory parameters for wheel"
    }
}
}]]
}

```

**Read snapshot data for specific properties of a thing - property** StringGeoValue**is of data type** GeoJSON

```

{
  "_id": "5A93B204ADCC4E8191FC3629541B2C48",
  "_thingType": [
    "core.automobiles:ABCXSeries"
  ],
  "value": [{
    "core.automobiles:ABCXSeries": [{
      "/StringCapability_T13_13": [{
        "StringGeoValue": {
          "_unitOfMeasure": "",
          "description": "",
          "_value": [78.47444, 17.37528],
          "_qualityCode": null,
          "_time": "2018-08-01T09:00:00.000Z"
        }
      }],
      "description": "",
      "_propertySetType": {
        "name": "core.automobiles:Wheel",
        "description": ""
      }
    }
  ]
}
}]]
}

```

## 15.2.5 Read Snapshot Data Within a Time Range for a Thing

Retrieves snapshot data of a thing within the specified time range

With this service, you can retrieve the latest value of properties for the specified thing ID. The service uses *fromTime* and *toTime*, to define the time range to retrieve the latest non-null value. In addition, the service retrieves the reference properties data of the specified thing ID. If you do not specify value or specify an empty value for the parameters *fromTime* and *toTime*, the service retrieves the first non-null value of properties. If the data is not available, the service returns null.

### i Note

The service retrieves the snapshot data for properties that support multiple languages in *Locale* specific language. If data is not available, the service returns null for the property.

## Request

URI: /v2/

Snapshot(thingId='<ID>',fromTime='<timestamp>',toTime='<timestamp>',dataCategory='<dataCategory>')

### i Note

The thing ID is mandatory to retrieve the snapshot details.

HTTP Method: *GET*

Permissions: <thing>.r

## Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <i>gzip</i> or <i>gzip,deflate</i> for this parameter, the system returns the compressed response with an additional response header CONTENT-ENCODING with value <i>gzip</i> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Parameters

Parameter	Required	Data Type	Description
ID	Yes	string	You can specify one of the following values: <ul style="list-style-type: none"><li>• Unique thing identifier generated by the system while creating a thing.</li><li>• Alternate thing identifier which is unique only within a specific tenant.</li></ul>
fromTime	Yes	timestamp	Time from which the snapshot details are retrieved for a non-null value of a property. The value for fromTime must correspond to UTC timestamp.

### i Note

If you do not specify a value or specify an empty value for this parameter, the service scans through the data available from the current time until it finds a non-null value. If the data is not available, the service returns null.

Parameter	Required	Data Type	Description
toTime	Yes	timestamp	Time up to which the snapshot details are retrieved for a non-null value of a property. The value for toTime must correspond to UTC timestamp.
dataCategory	Yes	enum	Filters the snapshot details based on data category enumeration values such as <a href="#">MasterData</a> , <a href="#">DerivedData</a> , <a href="#">TimeSeriesData</a> , <a href="#">Parameters</a> , and <a href="#">ReferencePropertyData</a> .
propertySetName	No	string	Property set type name
propertySetID	No	string	Property identifier

## Request Example

Read snapshot data of a thing within a specific time range

/v2/

```
Snapshot(thingId='503B8711B5D0445688E7FF7ACFC2177A',fromTime='2015-01-01T06:00:00Z',toTime='2015-05-01T06:00:00Z',dataCategory='TimeSeriesData')
```

Retrieves snapshot data for the specified thing ID within the time range: January 1, 2015 to May 1, 2015.

## Response

Format: [JSON](#)

### Response Properties

Property	Type	Description
_id	string	Internal identifier for a thing
_thingType	Array of thing type objects	Array of thing types

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [JSON](#)

Only media type [JSON](#) is supported. The [JSON](#) for this method has the following structure:

```
{
```



```

    "_id": "503B8711B5D0445688E7FF7ACFC2177A",
    "_thingType": [
      "core.automobiles:ABCXSeries"
    ],
    "_referenceProperty": [{
      "path": "/wheel_rr",
      "property": "Pressure",
      "_time": "2015-08-02T07:00:03Z",
      "com.sap.iot.core.UoM": "P",
      "com.sap.iot.core.LowerThreshold": 10,
      "com.sap.iot.core.UpperThreshold": 15,
      "com.sap.iot.core.LowLowerThreshold": 5,
      "com.sap.iot.core.UpperUpperThreshold": 20
    }, {
      "path": "/wheel_rr",
      "property": "Temperature",
      "_time": "2015-08-02T09:00:03Z",
      "com.sap.iot.core.UoM": "PA",
      "com.sap.iot.core.LowerThreshold": 10,
      "com.sap.iot.core.UpperThreshold": 20,
      "com.sap.iot.core.LowLowerThreshold": null,
      "com.sap.iot.core.UpperUpperThreshold": null
    }],
    "value": [{
      "core.automobiles:ABCXSeries": [{
        "/car": [{
          "BodyStyle": {
            "_value": "SUV",
            "_qualityCode": null,
            "_time": "2015-08-02T06:00:03Z",
            "_unitOfMeasure": ""
          }
        }, {
          "Class": {
            "_value": "1st",
            "_qualityCode": null,
            "_time": "2015-08-02T06:00:03Z",
            "_unitOfMeasure": ""
          }
        }, {
          "WheelBase": {
            "_value": "15",
            "_qualityCode": 9,
            "_time": "2015-08-02T06:00:03Z",
            "_unitOfMeasure": ""
          }
        }, {
          "RegistrationNumber": {
            "_value": "001",
            "_qualityCode": null,
            "_time": "2015-08-02T06:00:03Z",
            "_unitOfMeasure": ""
          }
        }, {
          "Colour": {
            "_value": "Black",
            "_qualityCode": null,
            "_time": "2015-08-02T06:00:03Z",
            "_unitOfMeasure": ""
          }
        }
      ]],
      "_propertySetType": {
        "name": "core.automobiles:Car",
        "description": "Sensory parameters for car"
      }
    }, {
      "/wheel_rr": [{
        "Pressure": {

```

```

        "_value": null,
        "_qualityCode": null,
        "_time": null,
        "_unitOfMeasure": "PA"
      }, {
        "Temperature": {
          "_value": null,
          "_qualityCode": null,
          "_time": null,
          "_unitOfMeasure": ""
        }
      }
    ],
    "_propertySetType": {
      "name": "core.automobiles:Wheel",
      "description": "Sensory parameters for wheel"
    }
  }
}
}
}
}

```

## 15.3 Time Series Cold Store

The Time Series Cold Store stores non-aggregated time series data for Things of tenant beyond the specified retention period of the tenant.

### Base URI

- **Formal description:** `http://<server address>[:<port number>][<path>]`
- **Example for a base URI in a cloud foundry environment:** `https://apiiot-coldstore.cfapps.eu10.hana.ondemand.com/Things`

### Methods

HTTP Method	Action	URI	Scopes
<i>PUT</i>	<a href="#">Create Time Series Data or Derived Data for a Thing [page 911]</a>	<code>/Things('&lt;thing ID&gt;')/&lt;thing type name&gt;/&lt;property set ID&gt;</code>	<code>&lt;coldstore&gt;.c</code>
<i>GET</i>	<a href="#">Read Time Series Data or Derived Data for a Thing [page 913]</a>	<code>/Things('&lt;thing ID&gt;')/&lt;thing type name&gt;/&lt;property set ID&gt;?timerange=&lt;fromTime - toTime&gt;</code>	<code>&lt;coldstore&gt;.r</code>

HTTP Method	Action	URI	Scopes
<a href="#">DELETE</a>	<a href="#">Delete Time Series Data or Derived Data [page 915]</a>	/Things('<thing ID>')/<property set ID>?time-range=<From time>-<To time> (OR) / Things('<thingID>')/<property set ID>? time-range=<fromTime – toTime>	<coldstore>.d

## 15.3.1 Create Time Series Data or Derived Data for a Thing

Create time series data or derived data for a thing

With this service, you can create data for the property sets of a thing that belongs to a specific thing type. This service supports data creation for the property sets of data categories [TimeSeriesData](#) and [DerivedData](#).

Using this service, you can send the request payload in [JSON](#) format or as a binary file in [zip](#) or [gzip](#) format. The complete request payload must be compressed in a single binary file and you cannot split the request payload in multiple files when using [zip](#) file format.

### Request

**URI:** /Things('<thing ID>')/<thing type name>/<property set ID>

#### i Note

- The thing type has to be a fully qualified name that is prefixed with the package name.
- The [\\_time](#) property is mandatory for the creation of data categories [TimeSeriesData](#) and [DerivedData](#). Following are the supported formats:
  - "/Date(<ticks>)/" <ticks> = number of milliseconds since midnight Jan 1, 1970  
For example, "/Date(1546322403148)/"
  - "YYYY-MM-DD HH:MM:SS[.fraction]"  
For example, "2015-08-01T09:00:00.000Z"
- You can only create time series data corresponding to a point in time that is outside the retention period specified for the tenant using this service or ingesting via SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (Internet of Things service). When you use this service, if the time stamp is within the retention period, an error occurs.
- Though the data creation time is not within the retention period, you cannot update existing data using this service for the properties of the property set type that are marked as personal data or sensitive data using the predefined annotations `com.sap.apptot.security:pii` and `com.sap.apptot.security:spi`.
- This service only supports request with business time value on or after 1970-01-01T00:00:00.000Z.
- Aggregates are not calculated for the time series data created using this service. Also, for data you created earlier, when you update it outside the retention period using this service, the aggregates are not updated.

HTTP Method: [PUT](#)

Permissions: <coldstore>.c

## Request Header Parameters

Header	Required	Description
content-Encoding	No	application/zip or application/gzip

**i Note**  
Indicates the content (request payload) to be processed is in application/zip or application/gzip file format.

## Request Parameters

Parameter	Required	Data Type	Description
thingID	Yes	String	Internal identifier for a thing.
thingTypeName	Yes	String	Name of a thing type.
propertySetID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

## Request Example

```
/Things('<thing ID>')/core.automobiles:ABCXSeries/wheel_fl
```

## Response

### Response Status and Error Codes

Code	Description
200	Data of category <a href="#">TimeSeriesData</a> or <a href="#">DerivedData</a> for the specified thing created successfully

## Payload

Format: [JSON](#)

Only media type *JSON* is supported. In addition, you can encode the request payload in *JSON* format to *zip* or *gzip* file format using the request header parameter `content-Encoding`. The *JSON* for this method has the structure as specified in the following example.

```
{
  "value": [{
    "time": "2015-08-01T09:00:00Z",
    "Pressure": "36",
    "Temperature": "100"
  }, {
    "time": "2015-08-02T06:00:03Z",
    "Pressure": "34",
    "Temperature": "120"
  }]
}
```

## 15.3.2 Read Time Series Data or Derived Data for a Thing

Retrieves property set data for the specified thing

With this service, you can retrieve the values of property sets of a thing that belongs to a specific thing type. You can request the values for a specific time range. The service retrieves the values of property sets of data categories *TimeSeriesData* or *DerivedData*.

### Request

**URI:** `/Things('<thing ID>')/<thing type name>/<property set ID>?timerange=<fromTime - toTime>`

#### i Note

- The thing type has to be a fully qualified name that is prefixed with the package.
- The service only retrieves data corresponding to a time range that is not within the retention period specified for the tenant. An error occurs if the time range is within the retention period.
- The time range you specify should not be greater than 48 hours.
- This service only supports request with business time value on or after 1970-01-01T00:00:00.000Z.

**HTTP Method:** *GET*

**Permissions:** `<coldstore>.r`

## Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header <code>CONTENT-ENCODING</code> with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Parameters

Parameter	Required	Data Type	Description
thingID	Yes	String	Internal identifier for a thing.
thingTypeName	Yes	String	Name of the thing type.
propertySetID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type
fromTime	Yes	timeStamp	Corresponds to the time value from which the time series data is retrieved for a non-null value of a property
toTime	Yes	timeStamp	Corresponds to the time value until which the time series data is retrieved for a non-null value of a property

## Request Example

```
/Things('<thing ID>')/core.automobiles:ABCXSeries/ wheel_fl?  
timerange=2015-08-01T09:00:00Z-2015-08-01T11:00:00Z
```

Retrieves property set data of the property set `wheel_fl` for the thing ID `core.automobiles:ABCXSeries` corresponding to a time range of two hours on August 1, 2015.

## Response

### Response Properties

Property	Type	Description
value	Array of thing data objects	An array of data objects (for example, time series data) for a thing or property set type.

## Response Status and Error Codes

Code	Description
200	Data for one or more things retrieved successfully

## Payload

```
{
  "value": [{
    "time": "2015-08-01T09:00:00Z",
    "Pressure": "36",
    "Temperature": "100"
  }, {
    "time": "2015-08-02T06:00:03Z",
    "Pressure": "34",
    "Temperature": "120"
  }]
}
```

## 15.3.3 Delete Time Series Data or Derived Data

Delete time series data or derived data for a thing

With this service, you can delete the time series data for the specified thing and property set type corresponding to the time range provided.

## Request

**URI:** /Things('<thing ID>')/<property set ID>?timerange=<From time>-<To time> (OR) / Things('<thingID>')/<property set ID>? timerange=<fromTime - toTime>

### i Note

- You must specify a time range to be able to delete the time series data.
- The service only deletes data corresponding to a time range that is not within the retention period specified for the tenant.
- This service only supports request with business time value on or after 1970-01-01T00:00:00.000Z.

**HTTP Method:** *DELETE*

**Permissions:** <coldstore>.d

## Request Parameters

Parameter	Required	Data Type	Description
thingID	Yes	String	Internal identifier for a thing.
propertySetID	Yes	String	Identifier of the property set.

## Request Example

```
/Things('<thingID>')/core.automobiles:ABCXSeries/car?  
timerange=2015-08-01T06:00:00Z-2015-08-01T08:00:00Z
```

## Response

### Response Status and Error Codes

Code	Description
200	Time series data beyond retention period deleted successfully.



# 16 Thing Time Series Data: OData Service

Retrieves time series data

## Overview

You can use this service to retrieve time series data of the property set type for the specified thing. You can request for the following information using this service:

- Single measurements for specific things for the specified timestamp
- Multiple measurements for specific things (or all things the user is allowed to access) for the specified time range
- Aggregated measurements for specific things (or all things the user is allowed to access) for the specified time range

### OData Version: 2.0

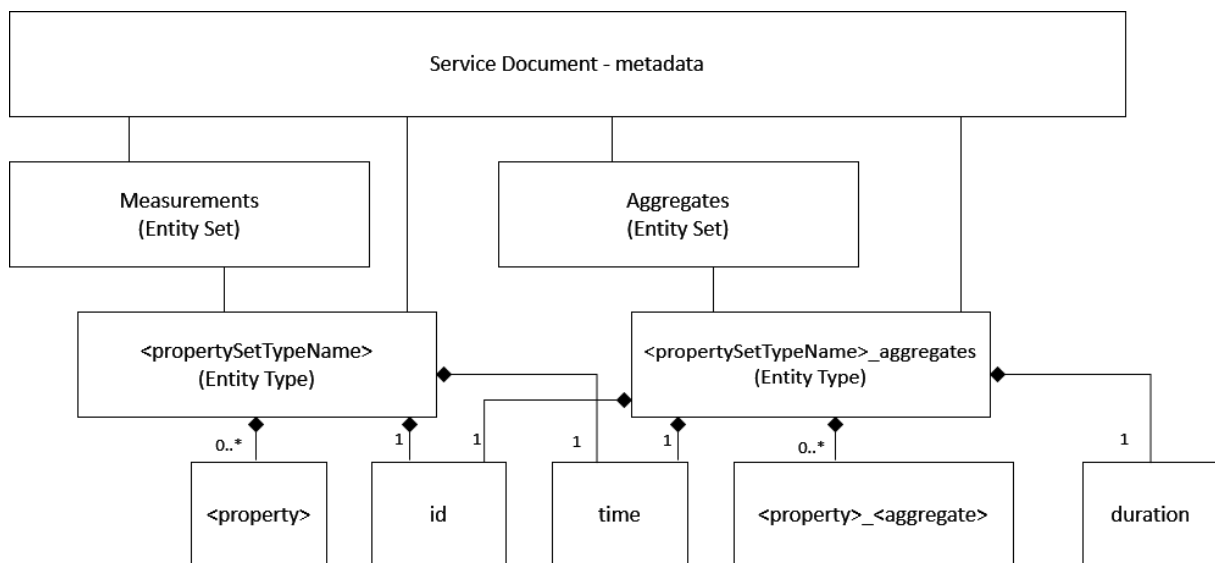
**Root URI:** `http://<server address>[:<port number>]/<property set type name>`

**Example for a Root URI in a cloud foundry environment:** `https://analytics-thing-sap.cfapps.eu10.hana.ondemand.com/<property set type name>`

**Permissions:** All scopes that already exist for the Thing Modeler apps are applicable to use this service. If you already have access to time series data in the thing model, you also have the rights to read analytical data. The same is applicable for administration rights.

## Entity Data Model

The following diagram shows the EDM of Time Series OData service:



**Service Metadata URI:** `http://<server address>[:<port number>]/<property set type name>/  
$metadata`

## Resources

Resource	Description	Path
<a href="#">Aggregates [page 921]</a>	Provides (read) access to aggregated thing time series data	<code>/aggregates</code>
<a href="#">Measurements [page 931]</a>	Provides (read) access to thing time series data	<code>/measurements</code>

## 16.1 Read Metadata

Retrieves metadata of time series OData service

### i Note

You cannot specify multiple property set types in the request URL.

## Request

**URI:** `/<property set type name>/$metadata`

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** All scopes that already exist for the Thing Modeler apps are applicable. If you already have access to time series data in the thing model, you also have the rights to read analytical data. The same is applicable for administration rights.

### Request Parameters

Parameter	Required	Data Type	Description
<code>property set type name</code>	Yes	String	Name of the property set type

### Request Example

`/<property set type name>/$metadata`

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

### Response Payload Example

The request URL `/core.automobiles.aggrePST/$metadata` retrieves the following response:

```
<EntityType Name="aggrePST">
  <Key>
    <PropertyRef Name="id"></PropertyRef>
    <PropertyRef Name="time"></PropertyRef>
  </Key>
  <Property Name="id" Type="Edm.String" sap:aggregation-
role="dimension" xmlns:sap="http://www.sap.com/Protocols/SAPData"></Property>
  <Property Name="time" Type="Edm.DateTime" sap:aggregation-
role="dimension" xmlns:sap="http://www.sap.com/Protocols/SAPData"></Property>
  <Property Name="pressure" Type="Edm.Int32" Nullable="true"></
Property>
  <Property Name="temperature" Type="Edm.Int32" Nullable="true"></
Property>
  <Property Name="_ThingType" Type="Edm.String" Nullable="true"></
Property>
  <Property Name="_NPST" Type="Edm.String" Nullable="true"></
Property>
  <Property Name="thingType" Type="Edm.String" Nullable="true"></
Property>
  <Property Name="path" Type="Edm.String" Nullable="true"></
Property>
</EntityType>
<EntityType Name="aggrePST_aggregates" sap:semantics="aggregate"
xmlns:sap="http://www.sap.com/Protocols/SAPData">
  <Key>
    <PropertyRef Name="_row_id"></PropertyRef>
  </Key>
  <Property Name="_row_id" Type="Edm.String"></Property>
  <Property Name="id" Type="Edm.String" sap:aggregation-
role="dimension" xmlns:sap="http://www.sap.com/Protocols/SAPData"></Property>
  <Property Name="time" Type="Edm.DateTime" sap:aggregation-
role="dimension" xmlns:sap="http://www.sap.com/Protocols/SAPData"></Property>
  <Property Name="duration" Type="Edm.String" Nullable="true"
sap:aggregation-role="dimension" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
  <Property Name="_ThingType" Type="Edm.String" Nullable="true"
sap:aggregation-role="dimension" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
  <Property Name="_NPST" Type="Edm.String" Nullable="true"
sap:aggregation-role="dimension" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
  <Property Name="thingType" Type="Edm.String" Nullable="true"
sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
  <Property Name="path" Type="Edm.String" Nullable="true"
sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
  <Property Name="pressure_MIN" Type="Edm.Int32" Nullable="true"
sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
  <Property Name="pressure_MAX" Type="Edm.Int32" Nullable="true"
sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
```

```

        <Property Name="pressure_AVG" Type="Edm.Double" Nullable="true"
sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
        <Property Name="pressure_SUM" Type="Edm.Double" Nullable="true"
sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
        <Property Name="pressure_COUNT" Type="Edm.Int64" Nullable="true"
sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
        <Property Name="pressure_STDDEV" Type="Edm.Double"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="pressure_FIRST" Type="Edm.Int32" Nullable="true"
sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
        <Property Name="pressure_LAST" Type="Edm.Int32" Nullable="true"
sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/Protocols/
SAPData"></Property>
        <Property Name="pressure_TFIRST" Type="Edm.DateTime"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="pressure_TLAST" Type="Edm.DateTime"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="pressure_TMIN" Type="Edm.DateTime"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="pressure_TMAX" Type="Edm.DateTime"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="pressure_PERCENT_GOOD" Type="Edm.Double"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_MIN" Type="Edm.Int32"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_MAX" Type="Edm.Int32"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_AVG" Type="Edm.Double"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_SUM" Type="Edm.Double"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_COUNT" Type="Edm.Int64"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_STDDEV" Type="Edm.Double"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_FIRST" Type="Edm.Int32"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_LAST" Type="Edm.Int32"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_TFIRST" Type="Edm.DateTime"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_TLAST" Type="Edm.DateTime"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_TMIN" Type="Edm.DateTime"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>

```

```

        <Property Name="temperature_TMAX" Type="Edm.DateTime"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
        <Property Name="temperature_PERCENT_GOOD" Type="Edm.Double"
Nullable="true" sap:aggregation-role="measure" xmlns:sap="http://www.sap.com/
Protocols/SAPData"></Property>
    </EntityType>
    <ComplexType Name="GeoLocation">
        <Property Name="latitude" Type="Edm.Single"></Property>
        <Property Name="longitude" Type="Edm.Single"></Property>
    </ComplexType>
    <EntityContainer Name="aggrePST_Container"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="measurements"
EntityType="core.automobiles.aggrePST"></EntitySet>
        <EntitySet Name="aggregates"
EntityType="core.automobiles.aggrePST_aggregates"></EntitySet>
    </EntityContainer>
</Schema>
<Schema Namespace="com.sap.apptot.timeseries" xmlns="http://
schemas.microsoft.com/ado/2008/09/edm">
    <ComplexType Name="GeoLocation">
        <Property Name="latitude" Type="Edm.Single"></Property>
        <Property Name="longitude" Type="Edm.Single"></Property>
    </ComplexType>
</Schema>
</edmx:DataServices>
</edmx:Edmx>

```

## Related Information

[Thing Time Series Data: OData Service \[page 917\]](#)

## 16.2 Aggregates

### Overview

The Aggregates resource provides access to aggregates of the time series data of things. The aggregates are calculated based on the following time intervals:

- Aggregates with granularity of less than 2 minutes are calculated precisely (on raw data)
- Aggregates with granularity between 2 minutes and 1 hour size are calculated based on 2-minute pre-calculated aggregates
- Aggregates with granularity between 1 hour and 1 day (24 hours) are calculated based on 1-hour pre-calculated aggregates
- Aggregates with granularity between 1 day (24 hours) and 1 week (7 days) are calculated on 1-day pre-calculated aggregates

- Aggregates with granularity of more than 1 week (7 days) are calculated on 7-days pre-calculated aggregates

For example, if you request for aggregates for 7 minutes, the service returns the multiple of 2-minute pre-calculated aggregates which is 6 minutes.

**Resource Path:** `http://<server address>[:<port number>]/<property set type name>/aggregates`

**Permissions:** All scopes that already exist for the Thing Modeler apps are applicable. If you already have access to time series data in the thing model, you also have the rights to read analytical data. The same is applicable for administration rights.

## Operations

### CRUD Operations

HTTP Method	Operation	URI
<a href="#">GET</a>	<a href="#">Read Aggregates of Measurements</a> <a href="#">[page 922]</a>	<code>/&lt;property set type name&gt;/aggregates? \$filter=&lt;&gt;</code>

## Related Information

[Thing Time Series Data: OData Service \[page 917\]](#)

### 16.2.1 Read Aggregates of Measurements

Retrieves aggregates of measurements

With this method, you can retrieve aggregates of measurements for specified things or for all things (you have authorization to access). You can retrieve data for a specific time range. The method retrieves a maximum of 1000 result entries per response. You can navigate to the next set of results, if available, using the **\_\_next** link at the end in the response payload.

You must use the `$filter` query parameter on the `time` property to define the time range to filter data. You can use the property `thing ID` to retrieve aggregates for the specified list of things. If you do not specify the thing IDs, the service retrieves aggregates for all things (you have authorization to access).

#### **i** Note

The system only supports a maximum storage capacity of 10GB to store aggregates per property set type.

## Properties

### Dimension Property

The service does not allow you to aggregate the dimension properties and the service retains the original names without additional suffix for the dimension properties.

The aggregate entity type in the `$metadata` document contains multiple properties for each non-dimensional property of a property set type, one for each supported aggregate type. The resulting property names have suffix like `_COUNT` that corresponds to the following aggregate type:

- `COUNT, FIRST, LAST, TFIRST, TLAST` - supported for all base-types
- `MIN, MAX, TMIN, TMAX` - supported for all comparable base-types
- `SUM, AVG, STDDEV` (sum, average, and standard deviation) - supported for numeric base-types

### Note

For more information about how to define a property of type *Dimension*, see [Create a Property Set](#).

## Duration Property

In addition to the standard properties `id` (thing ID) and `time`, the entity types contain the duration property that defines the "size" of the aggregates. This property only accepts data of type *String* and supports one of the following formats:

- `[#D][T[#H][#M][#[.#]S]]` for concrete time periods in Days (24h), Hours, Minutes and (Sub-)Seconds.
- `[#Y][#M]` for logical time periods Years and Months.

The query parameter `$filter` only supports filter operator `eq` for the duration property.

## Examples

The following are few examples to explain how aggregates are retrieved based on the duration.

- `duration=T0.5S`, retrieves 500 milliseconds aggregates (based on single measurements)
- `duration=T10S`, retrieves 10 seconds aggregates (based on single measurements)
- `duration=T10M`, retrieves 10 minutes aggregates (based on 2-minute preaggregates)
- `duration=T12H`, retrieves 12 hours aggregates (based on hourly preaggregates)
- `duration=2D`, retrieves 2 days or 48 hours aggregates (based on hourly aggregates)
- `duration=14D`, retrieves 14 days aggregates (based on weekly preaggregates)
- `duration=3M`, retrieves 3 months aggregates (based on daily preaggregates)
- `duration=1Y`, retrieves 1 year aggregates (based on daily preaggregates)

Due to limited granularities, the following durations might yield deviating results:

- `duration=T15M`, retrieves 14 minutes aggregates (based on 2-minute preaggregates)
- `duration=T1H30M`, retrieves 1 hour aggregates (based on hourly preaggregates)
- `duration=4DT12H10M30S`, retrieves 4 days aggregates (based on daily preaggregates)
- `duration=15D`, retrieves 14 days or 2 weeks aggregates (based on weekly preaggregates)

## Grouping of Aggregates

You can group the aggregates based on the following properties:

- Dimension property
- `id` property
- `time` (duration) property

You can group the aggregates based on the `id` and dimension properties, by specifying the properties in the `$select` query parameter.

You can group the aggregates based on the `time` property, by specifying the duration property in `$filter` query parameter. You use the duration property to define the granularity.

If you do not specify `$select`, the service retrieves all supported aggregates for all available structure fields. The service groups the aggregates based on the properties `id` and dimension fields.

### Single Aggregates or Snapshot

If you do not specify the duration in `$filter` query parameter, the service retrieves single result per group. The service calculates the aggregates for the entire time range specified in the query.

If you specify the `LAST` aggregate type along with a time range until the current time, the service retrieves the latest values of the selected properties.

## Request

**URI:** `/<property set type name>/aggregates?$filter=id eq '<thing ID>' and time ge datetime'<date time>' and time lt datetime'<date time>' and duration eq 'T60M'`

#### Note

`$filter` on time property is mandatory in the request URL. `$filter` on other supported properties are optional.

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** All scopes that already exist for the Thing Modeler apps are applicable. If you already have access to time series data in the thing model, you also have the rights to read analytical data. The same is applicable for administration rights.

### Request Parameters

Parameter	Required	Data Type	Description
<code>property set type name</code>	Yes	String	Name of the property set type



## Query Request Parameters

Parameter	Required	Data Type	Description
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are <code>id</code>, <code>time</code>, <code>duration</code>, <code>dimension</code> properties, <code>_row_id</code>, and each aggregate of fields of the property set type assigned to the thing (for example, <code>&lt;field&gt;_&lt;aggregate&gt;</code>).</p> <p>For productive use and better results, we recommended you to always include all the valid fields in the <code>\$select</code> query.</p>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; <code>time</code> is the only field supported.</p> <p>To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</p>
\$top	No	Integer	Number of records to include in the result set
\$filter	Yes	String	Filter condition to be applied; valid fields are <code>id</code> , <code>time</code> , <code>duration</code> , and each aggregate of fields of the property set type assigned to the thing. The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> . However, you can only use the filter operator <a href="#">eq</a> to compare the field <code>id</code> .

## Request Example

```
/<property set type name>/aggregates?$filter=id eq 'id1' and time ge  
datetime'2017-01-01T00:00:00' and time lt datetime'2017-03-01T00:00:00'
```

Retrieves aggregates of measurements for the thing `id1` between the time range `2017-01-01T00:00:00` and `2017-03-01T00:00:00`.

```
/<property set type name>/aggregates?  
$select=id,time,duration,dimension1,dimension2,sensordata_MAX&$filter=id eq 'id1'  
and time ge datetime'2017-01-01T00:00:00' and time lt datetime'2017-01-01T01:00:00'  
and duration eq 'T60M'&$orderby=time desc&$top=10
```

Retrieves aggregates of measurements for the thing `id1` between the time range `2017-01-01T00:00:00` and `2017-01-01T01:00:00` and for a duration of 60 minutes, grouped by dimension properties `dimension1` and `dimension2`. The result set contains the selected fields `id`, `time`, `duration`, and the dimension properties `dimension1`, `dimension2`, and the aggregate `sensordata_MAX`. The retrieved entries are sorted by `time` in descending order.

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully (includes empty result set)

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as illustrated in the following examples.

### Example 1

The request `/core.automobiles:aggrePST/aggregates?$filter=time ge datetime'2018-01-20T00:00:00' and time lt datetime'2019-01-25T00:00:00'` retrieves the following result:

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://analytics-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2018-01-20T00%3A00%3A00Z%2Cduration%3DALL%28T8880H%29%2Cid%3DA2E9050BAFD7476B97AC4CCA0E309A35%2CthingType%3Dcore.automobiles%3AaggreTT%2Cpath%3Daggrepropset')",
        "uri": "https://analytics-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2018-01-20T00%3A00%3A00Z%2Cduration%3DALL%28T8880H%29%2Cid%3DA2E9050BAFD7476B97AC4CCA0E309A35%2CthingType%3Dcore.automobiles%3AaggreTT%2Cpath%3Daggrepropset')",
        "type": "core.automobiles.aggrePST_aggregates"
      },
      "row_id":
      "time=2018-01-20T00:00:00Z,duration=ALL(T8880H),id=A2E9050BAFD7476B97AC4CCA0E309A35,thingType=core.automobiles:aggreTT,path=aggrepropset",
      "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
      "time": "/Date(1516406400000)/",
      "duration": "ALL(T8880H)",
      "_ThingType": "core.automobiles:aggreTT",
      "NPST": "aggrepropset",
      "thingType": "core.automobiles:aggreTT",
      "path": "aggrepropset",
      "pressure_MIN": 20,
      "pressure_MAX": 274,
      "pressure_AVG": "132.0",
      "pressure_SUM": "792.0",
    }],
  }
}
```

```

        "pressure_COUNT": "6",
        "pressure_STDDEV": "111.21750461745368",
        "pressure_FIRST": 20,
        "pressure_LAST": 274,
        "pressure_TFIRST": "/Date(1539993600000)/",
        "pressure_TLAST": "/Date(1548115200000)/",
        "pressure_TMIN": "/Date(1539993600000)/",
        "pressure_TMAX": "/Date(1548115200000)/",
        "pressure_PERCENT_GOOD": "100.0",
        "temperature_MIN": 32,
        "temperature_MAX": 368,
        "temperature_AVG": "187.66666666666666",
        "temperature_SUM": "1126.0",
        "temperature_COUNT": "6",
        "temperature_STDDEV": "154.45999122390526",
        "temperature_FIRST": 34,
        "temperature_LAST": 368,
        "temperature_TFIRST": "/Date(1539993600000)/",
        "temperature_TLAST": "/Date(1548115200000)/",
        "temperature_TMIN": "/Date(1540166400000)/",
        "temperature_TMAX": "/Date(1548115200000)/",
        "temperature_PERCENT_GOOD": "100.0"
    },
    {
        "_metadata": {
            "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2018-01-20T00%3A00%3A00Z
%2Cduration%3DALL%28T8880H%29%2Cid%3D39F18733B0734903A67F7A1D9B9791A4%2CthingType
%3Dcore.automobiles%3AaggreTT2%2Cpath%3Daggrepropset2')",
            "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2018-01-20T00%3A00%3A00Z
%2Cduration%3DALL%28T8880H%29%2Cid%3D39F18733B0734903A67F7A1D9B9791A4%2CthingType
%3Dcore.automobiles%3AaggreTT2%2Cpath%3Daggrepropset2')",
            "type": "core.automobiles.aggrePST_aggregates"
        },
        "row_id":
"time=2018-01-20T00:00:00Z,duration=ALL(T8880H),id=39F18733B0734903A67F7A1D9B9791
A4,thingType=core.automobiles:aggreTT2,path=aggrepropset2",
        "id": "39F18733B0734903A67F7A1D9B9791A4",
        "time": "/Date(1516406400000)/",
        "duration": "ALL(T8880H)",
        "_ThingType": "core.automobiles:aggreTT2",
        "_NPST": "aggrepropset2",
        "thingType": "core.automobiles:aggreTT2",
        "path": "aggrepropset2",
        "pressure_MIN": 20,
        "pressure_MAX": 874,
        "pressure_AVG": "238.66666666666666",
        "pressure_SUM": "1432.0",
        "pressure_COUNT": "6",
        "pressure_STDDEV": "294.8302713238396",
        "pressure_FIRST": 120,
        "pressure_LAST": 74,
        "pressure_TFIRST": "/Date(1546560000000)/",
        "pressure_TLAST": "/Date(1548028800000)/",
        "pressure_TMIN": "/Date(1547856000000)/",
        "pressure_TMAX": "/Date(1547769600000)/",
        "pressure_PERCENT_GOOD": "100.0",
        "temperature_MIN": 8,
        "temperature_MAX": 214,
        "temperature_AVG": "81.33333333333333",
        "temperature_SUM": "488.0",
        "temperature_COUNT": "6",
        "temperature_STDDEV": "74.06903686576614",
        "temperature_FIRST": 214,
        "temperature_LAST": 8,
        "temperature_TFIRST": "/Date(1546560000000)/",
        "temperature_TLAST": "/Date(1548028800000)/",
    }

```

```

        "temperature_TMIN": "/Date(1548028800000)/",
        "temperature_TMAX": "/Date(1546560000000)/",
        "temperature_PERCENT_GOOD": "100.0"
    }
}
]
}
}

```

## Example 2

The request `/core.automobiles:aggrePST/aggregates?$filter=id eq '39F18733B0734903A67F7A1D9B9791A4' and time ge datetime'2018-01-20T00:00:00' and time lt datetime'2019-01-25T00:00:00'&$select=id,time,duration,pressure_MIN,pressure_MAX,thingType,path and duration eq 'T60M'&$orderby=time desc&$top=5` retrieves the following result:

```

{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://analytics-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2019-01-21T00%3A00%3A00Z%2Cduration%3DT1H%2Cid%3D39F18733B0734903A67F7A1D9B9791A4')",
        "uri": "https://analytics-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2019-01-21T00%3A00%3A00Z%2Cduration%3DT1H%2Cid%3D39F18733B0734903A67F7A1D9B9791A4')",
        "type": "core.automobiles.aggrePST_aggregates"
      },
      "id": "39F18733B0734903A67F7A1D9B9791A4",
      "time": "/Date(1548028800000)/",
      "duration": "T1H",
      "_ThingType": "core.automobiles:aggreTT2",
      "NPST": "aggrepropset2",
      "thingType": "core.automobiles:aggreTT2",
      "path": "aggrepropset2",
      "pressure_MIN": 74,
      "pressure_MAX": 74
    },
    {
      "__metadata": {
        "id": "https://analytics-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2019-01-20T00%3A00%3A00Z%2Cduration%3DT1H%2Cid%3D39F18733B0734903A67F7A1D9B9791A4')",
        "uri": "https://analytics-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2019-01-20T00%3A00%3A00Z%2Cduration%3DT1H%2Cid%3D39F18733B0734903A67F7A1D9B9791A4')",
        "type": "core.automobiles.aggrePST_aggregates"
      },
      "id": "39F18733B0734903A67F7A1D9B9791A4",
      "time": "/Date(1547942400000)/",
      "duration": "T1H",
      "_ThingType": "core.automobiles:aggreTT2",
      "NPST": "aggrepropset2",
      "thingType": "core.automobiles:aggreTT2",
      "path": "aggrepropset2",
      "pressure_MIN": 72,
      "pressure_MAX": 72
    },
    {
      "__metadata": {
        "id": "https://analytics-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2019-01-19T00%3A00%3A00Z%2Cduration%3DT1H%2Cid%3D39F18733B0734903A67F7A1D9B9791A4')",
        "uri": "https://analytics-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2019-01-19T00%3A00%3A00Z%2Cduration%3DT1H%2Cid%3D39F18733B0734903A67F7A1D9B9791A4')",

```

```

        "type": "core.automobiles.aggrePST_aggregates"
    },
    "id": "39F18733B0734903A67F7A1D9B9791A4",
    "time": "/Date(1547856000000)/",
    "duration": "T1H",
    "_ThingType": "core.automobiles:aggreTT2",
    "_NPST": "aggrepropset2",
    "thingType": "core.automobiles:aggreTT2",
    "path": "aggrepropset2",
    "pressure_MIN": 20,
    "pressure_MAX": 20
},
{
    "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2019-01-18T00%3A00%3A00Z
%2Cduration%3DT1H%2Cid%3D39F18733B0734903A67F7A1D9B9791A4')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2019-01-18T00%3A00%3A00Z
%2Cduration%3DT1H%2Cid%3D39F18733B0734903A67F7A1D9B9791A4')",
        "type": "core.automobiles.aggrePST_aggregates"
    },
    "id": "39F18733B0734903A67F7A1D9B9791A4",
    "time": "/Date(1547769600000)/",
    "duration": "T1H",
    "_ThingType": "core.automobiles:aggreTT2",
    "_NPST": "aggrepropset2",
    "thingType": "core.automobiles:aggreTT2",
    "path": "aggrepropset2",
    "pressure_MIN": 874,
    "pressure_MAX": 874
},
{
    "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2019-01-06T00%3A00%3A00Z
%2Cduration%3DT1H%2Cid%3D39F18733B0734903A67F7A1D9B9791A4')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2019-01-06T00%3A00%3A00Z
%2Cduration%3DT1H%2Cid%3D39F18733B0734903A67F7A1D9B9791A4')",
        "type": "core.automobiles.aggrePST_aggregates"
    },
    "id": "39F18733B0734903A67F7A1D9B9791A4",
    "time": "/Date(1546732800000)/",
    "duration": "T1H",
    "_ThingType": "core.automobiles:aggreTT2",
    "_NPST": "aggrepropset2",
    "thingType": "core.automobiles:aggreTT2",
    "path": "aggrepropset2",
    "pressure_MIN": 272,
    "pressure_MAX": 272
}
    ]
}
}

```

### Example 3

The request `/core.automobiles:aggrePST/aggregates?$filter=id eq 'A2E9050BAFD7476B97AC4CCA0E309A35' and time ge datetime'2018-01-20T00:00:00' and time lt datetime'2019-01-25T00:00:00'` retrieves the following result:

```

{
  "d": {
    "results": [{
      "__metadata": {

```

```

        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2018-01-20T00%3A00%3A00Z
%2Cduration%3DALL%28T8880H%29%2Cid%3DA2E9050BAFD7476B97AC4CCA0E309A35%2CthingType
%3Dcore.automobiles%3AaggreTT%2Cpath%3Daggrepropset')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/aggregates('time%3D2018-01-20T00%3A00%3A00Z
%2Cduration%3DALL%28T8880H%29%2Cid%3DA2E9050BAFD7476B97AC4CCA0E309A35%2CthingType
%3Dcore.automobiles%3AaggreTT%2Cpath%3Daggrepropset')",
        "type": "iotae.citest1.core.taggregates.aggrePST_aggregates"
    },
    " row id":
"time=2018-01-20T00:00:00Z,duration=ALL(T8880H),id=A2E9050BAFD7476B97AC4CCA0E309A
35,thingType=core.automobiles:aggreTT,path=aggrepropset",
    "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
    "time": "/Date(1516406400000)/",
    "duration": "ALL(T8880H)",
    "_ThingType": "core.automobiles:aggreTT",
    "_NPST": "aggrepropset",
    "thingType": "core.automobiles:aggreTT",
    "path": "aggrepropset",
    "pressure_MIN": 20,
    "pressure_MAX": 274,
    "pressure_AVG": "132.0",
    "pressure_SUM": "792.0",
    "pressure_COUNT": "6",
    "pressure_STDDEV": "111.21750461745368",
    "pressure_FIRST": 20,
    "pressure_LAST": 274,
    "pressure_TFIRST": "/Date(1539993600000)/",
    "pressure_TLAST": "/Date(1548115200000)/",
    "pressure_TMIN": "/Date(1539993600000)/",
    "pressure_TMAX": "/Date(1548115200000)/",
    "pressure_PERCENT_GOOD": "100.0",
    "temperature_MIN": 32,
    "temperature_MAX": 368,
    "temperature_AVG": "187.66666666666666",
    "temperature_SUM": "1126.0",
    "temperature_COUNT": "6",
    "temperature_STDDEV": "154.45999122390526",
    "temperature_FIRST": 34,
    "temperature_LAST": 368,
    "temperature_TFIRST": "/Date(1539993600000)/",
    "temperature_TLAST": "/Date(1548115200000)/",
    "temperature_TMIN": "/Date(1540166400000)/",
    "temperature_TMAX": "/Date(1548115200000)/",
    "temperature_PERCENT_GOOD": "100.0"
}
    }
}

```

## Related Information

[Aggregates \[page 921\]](#)

## 16.3 Measurements

Retrieves measurements

With this service, you can retrieve single measurements for specific things for the specified timestamp. You can also retrieve multiple measurements for specific things (or all things the user is allowed to access) for the specified time range.

**Resource Path:** `http://<server address>[:<port number>]/<property set type name>/measurements`

**Permissions:** All scopes that already exist for the Thing Modeler apps are applicable. If you already have access to time series data in the thing model, you also have the rights to read analytical data. The same is applicable for administration rights.

### Operations

#### CRUD Operations

HTTP Method	Operation	URI
GET	<a href="#">Read Single Measurement [page 931]</a>	<code>/&lt;property set type name&gt;/measurements(id='&lt;thing ID&gt;', time=datetime'&lt;date time&gt;')</code>
GET	<a href="#">Read Multiple Measurements [page 933]</a>	<code>/&lt;property set type name&gt;/measurements?\$filter=id eq '&lt;thing ID1&gt;' or id eq '&lt;thing ID 2&gt;' and time ge datetime'&lt;date time&gt;' and time lt datetime'&lt;date time&gt;'</code>

### Related Information

[Thing Time Series Data: OData Service \[page 917\]](#)

#### 16.3.1 Read Single Measurement

Retrieves single measurement

With this service, you can retrieve single measurement for a property set type used by the specified thing for a specific timestamp.

You can use the standard properties `id` (thing ID) and `time` to retrieve the single measurement value.

## Request

**URI:** `/<property set type name>/measurements(id='<thing ID>', time=datetime'<date time>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** All scopes that already exist for the Thing Modeler apps are applicable. If you already have access to time series data in the thing model, you also have the rights to read analytical data. The same is applicable for administration rights.

### Request Parameters

Parameter	Required	Data Type	Description
id	Yes	String	Unique identifier of the thing
time	Yes	DateTime	Corresponds to UTC date and time (with seconds precision)
property set type name	Yes	String	Name of the property set type

### Request Example

```
<property set type name>/measurements(id='id1',time=datetime'2017-01-01T00:00:00')
```

Retrieves single measurement for the thing `id1` for the timestamp `2017-01-01T00:00:00`.

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully (includes empty result set)
404	No data found for the specified property set type, thing ID, and time range

## Payload

**Format:** [JSON](#)

```
/core.automobiles:aggrePST/  
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-20T00:00:00')
```

```
{  
  "d": {  
    "__metadata": {
```



```

      "id": "https://analytics-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-20T00%3A
00%3A00')",
      "uri": "https://analytics-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-20T00%3A
00%3A00')",
      "type": "core.automobiles.aggrePST"
    },
    "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
    "time": "/Date(1539993600000)/",
    "pressure": 20,
    "temperature": 34,
    "_ThingType": "core.automobiles:aggreTT",
    "_NPST": "aggrepropset",
    "thingType": "core.automobiles:aggreTT",
    "path": "aggrepropset"
  }
}

```

## Related Information

[Measurements \[page 931\]](#)

## 16.3.2 Read Multiple Measurements

Retrieves multiple measurements

With this method, you can retrieve multiple measurements for specific things (or all things the user is allowed to access) for the specified time range. The method retrieves a maximum of 1000 result entries per response. You can use the **\_\_next** link to access subsequent results, if available.

You must use the `$filter` query parameter on the `time` property to define the time range to filter data. You can use the property `thing ID` to retrieve multiple measurements for the specified list of things. If you do not specify the thing IDs, the service retrieves multiple measurements for all things you have authorized access

## Request

**URI:** `/<property set type name>/measurements?$filter=id eq '<thing ID1>' or id eq '<thing ID 2>' and time ge datetime'<date time>' and time lt datetime'<date time>'`

### Note

`$filter` on time property is mandatory in the request URL. `$filter` on other supported properties is optional.

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** All scopes that already exist for the Thing Modeler apps are applicable. If you already have access to time series data in the thing model, you also have the rights to read analytical data. The same is applicable for administration rights.

## Request Parameters

Parameter	Required	Data Type	Description
property set type name	Yes	String	Name of the property set type

## Query Request Parameters

Parameter	Required	Data Type	Description
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are <code>id</code> , <code>time</code> , and all fields of the property set type assigned to the thing.
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; <code>time</code> is the only field supported.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$top	No	Integer	Number of records to include in the result set.
\$filter	Yes	String	Filter condition to be applied; valid fields are <code>id</code> , <code>time</code> , and all fields of the property set type assigned to the thing. The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> . However, you can only use the filter operator <a href="#">eq</a> to compare the field <code>id</code> .

## Request Example

```
/<property set type name>/measurements?$filter=id eq 'id1' or id eq 'id2' and time ge datetime'<2017-01-01T00:00:00>' and time lt datetime'<2018-01-01T00:00:00>'
```

Retrieves multiple measurements for the specified things `id1` and `id2` between the time range 2017-01-01T00:00:00 and 2018-01-01T00:00:00.

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully (includes empty result set)

## Payload

Format: [JSON](#)

Media type supported are [JSON](#) and [XML](#). The [JSON](#) format for this method has the structure as illustrated in the following examples.

### Example 1

The request `/core.automobiles:aggrePST/measurements?$filter=id eq 'A2E9050BAFD7476B97AC4CCA0E309A35' and time ge datetime'2018-01-20T00:00:00' and time lt datetime'2019-01-25T00:00:00'` retrieves the following result:

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-22T00%3A
00%3A00')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-22T00%3A
00%3A00')",
        "type": "core.automobiles.aggrePST"
      },
      "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
      "time": "/Date(1548115200000)/",
      "pressure": 274,
      "temperature": 368,
      "_ThingType": "core.automobiles:aggreTT",
      "_NPST": "aggrepropset",
      "thingType": "core.automobiles:aggreTT",
      "path": "aggrepropset"
    },
    {
      "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-05T00%3A
00%3A00')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-05T00%3A
00%3A00')",
        "type": "core.automobiles.aggrePST"
      },
      "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
      "time": "/Date(1546646400000)/",
      "pressure": 232,
```

```

        "temperature": 342,
        "_ThingType": "core.automobiles:aggreTT",
        "_NPST": "aggrepropset",
        "thingType": "core.automobiles:aggreTT",
        "path": "aggrepropset"
    },
    {
        "__metadata": {
            "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-02T00%3A
00%3A00')",
            "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-02T00%3A
00%3A00')",
            "type": "core.automobiles:aggrePST"
        },
        "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
        "time": "/Date(1546387200000)/",
        "pressure": 220,
        "temperature": 314,
        "_ThingType": "core.automobiles:aggreTT",
        "_NPST": "aggrepropset",
        "thingType": "core.automobiles:aggreTT",
        "path": "aggrepropset"
    },
    {
        "__metadata": {
            "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-24T00%3A
00%3A00')",
            "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-24T00%3A
00%3A00')",
            "type": "core.automobiles:aggrePST"
        },
        "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
        "time": "/Date(1540339200000)/",
        "pressure": 24,
        "temperature": 36,
        "_ThingType": "core.automobiles:aggreTT",
        "_NPST": "aggrepropset",
        "thingType": "core.automobiles:aggreTT",
        "path": "aggrepropset"
    },
    {
        "__metadata": {
            "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-22T00%3A
00%3A00')",
            "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-22T00%3A
00%3A00')",
            "type": "core.automobiles:aggrePST"
        },
        "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
        "time": "/Date(1540166400000)/",
        "pressure": 22,
        "temperature": 32,
        "_ThingType": "core.automobiles:aggreTT",
        "_NPST": "aggrepropset",
        "thingType": "core.automobiles:aggreTT",
        "path": "aggrepropset"
    }

```

```

    },
    {
      "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-20T00%3A
00%3A00')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-20T00%3A
00%3A00')",
        "type": "core.automobiles.aggrePST"
      },
      "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
      "time": "/Date(1539993600000)/",
      "pressure": 20,
      "temperature": 34,
      "_ThingType": "core.automobiles.aggreTT",
      "NPST": "aggrepropset",
      "thingType": "core.automobiles.aggreTT",
      "path": "aggrepropset"
    }
  ]
}

```

## Example 2

The request `/core.automobiles:aggrePST/measurements?$filter=time ge datetime'2018-01-20T00:00:00' and time lt datetime'2019-01-25T00:00:00'` retrieves the following result:

```

{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-22T00%3A
00%3A00')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-22T00%3A
00%3A00')",
        "type": "core.automobiles.aggrePST"
      },
      "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
      "time": "/Date(1548115200000)/",
      "pressure": 274,
      "temperature": 368,
      "_ThingType": "core.automobiles.aggreTT",
      "NPST": "aggrepropset",
      "thingType": "core.automobiles.aggreTT",
      "path": "aggrepropset"
    },
    {
      "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-21T00%3A
00%3A00')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-21T00%3A
00%3A00')",
        "type": "core.automobiles.aggrePST"
      }
    }
  ]
}

```

```

    },
    "id": "39F18733B0734903A67F7A1D9B9791A4",
    "time": "/Date(1548028800000)/",
    "pressure": 74,
    "temperature": 8,
    "_ThingType": "core.automobiles:aggreTT2",
    "_NPST": "aggrepropset2",
    "thingType": "core.automobiles:aggreTT2",
    "path": "aggrepropset2"
  },
  {
    "__metadata": {
      "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-20T00%3A
00%3A00')",
      "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-20T00%3A
00%3A00')",
      "type": "core.automobiles:aggrePST"
    },
    "id": "39F18733B0734903A67F7A1D9B9791A4",
    "time": "/Date(1547942400000)/",
    "pressure": 72,
    "temperature": 42,
    "_ThingType": "core.automobiles:aggreTT2",
    "_NPST": "aggrepropset2",
    "thingType": "core.automobiles:aggreTT2",
    "path": "aggrepropset2"
  },
  {
    "__metadata": {
      "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-19T00%3A
00%3A00')",
      "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-19T00%3A
00%3A00')",
      "type": "core.automobiles:aggrePST"
    },
    "id": "39F18733B0734903A67F7A1D9B9791A4",
    "time": "/Date(1547856000000)/",
    "pressure": 20,
    "temperature": 14,
    "_ThingType": "core.automobiles:aggreTT2",
    "_NPST": "aggrepropset2",
    "thingType": "core.automobiles:aggreTT2",
    "path": "aggrepropset2"
  },
  {
    "__metadata": {
      "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-18T00%3A
00%3A00')",
      "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-18T00%3A
00%3A00')",
      "type": "core.automobiles:aggrePST"
    },
    "id": "39F18733B0734903A67F7A1D9B9791A4",
    "time": "/Date(1547769600000)/",
    "pressure": 874,
    "temperature": 68,

```

```

        "_ThingType": "core.automobiles:aggreTT2",
        "_NPST": "aggrepropset2",
        "thingType": "core.automobiles:aggreTT2",
        "path": "aggrepropset2"
    },
    {
        "__metadata": {
            "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-06T00%3A
00%3A00')",
            "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-06T00%3A
00%3A00')",
            "type": "core.automobiles:aggrePST"
        },
        "id": "39F18733B0734903A67F7A1D9B9791A4",
        "time": "/Date(1546732800000)/",
        "pressure": 272,
        "temperature": 142,
        "_ThingType": "core.automobiles:aggreTT2",
        "_NPST": "aggrepropset2",
        "thingType": "core.automobiles:aggreTT2",
        "path": "aggrepropset2"
    },
    {
        "__metadata": {
            "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-05T00%3A
00%3A00')",
            "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-05T00%3A
00%3A00')",
            "type": "core.automobiles:aggrePST"
        },
        "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
        "time": "/Date(1546646400000)/",
        "pressure": 232,
        "temperature": 342,
        "_ThingType": "core.automobiles:aggreTT",
        "_NPST": "aggrepropset",
        "thingType": "core.automobiles:aggreTT",
        "path": "aggrepropset"
    },
    {
        "__metadata": {
            "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-04T00%3A
00%3A00')",
            "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='39F18733B0734903A67F7A1D9B9791A4',time=datetime'2019-01-04T00%3A
00%3A00')",
            "type": "core.automobiles:aggrePST"
        },
        "id": "39F18733B0734903A67F7A1D9B9791A4",
        "time": "/Date(1546560000000)/",
        "pressure": 120,
        "temperature": 214,
        "_ThingType": "core.automobiles:aggreTT2",
        "_NPST": "aggrepropset2",
        "thingType": "core.automobiles:aggreTT2",
        "path": "aggrepropset2"
    },

```

```

    {
      "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-02T00%3A
00%3A00')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2019-01-02T00%3A
00%3A00')",
        "type": "core.automobiles.aggrePST"
      },
      "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
      "time": "/Date(1546387200000)/",
      "pressure": 220,
      "temperature": 314,
      "_ThingType": "core.automobiles.aggreTT",
      "_NPST": "aggrepropset",
      "thingType": "core.automobiles.aggreTT",
      "path": "aggrepropset"
    },
    {
      "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-24T00%3A
00%3A00')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-24T00%3A
00%3A00')",
        "type": "core.automobiles.aggrePST"
      },
      "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
      "time": "/Date(1540339200000)/",
      "pressure": 24,
      "temperature": 36,
      "_ThingType": "core.automobiles.aggreTT",
      "_NPST": "aggrepropset",
      "thingType": "core.automobiles.aggreTT",
      "path": "aggrepropset"
    },
    {
      "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-22T00%3A
00%3A00')",
        "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-22T00%3A
00%3A00')",
        "type": "core.automobiles.aggrePST"
      },
      "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
      "time": "/Date(1540166400000)/",
      "pressure": 22,
      "temperature": 32,
      "_ThingType": "core.automobiles.aggreTT",
      "_NPST": "aggrepropset",
      "thingType": "core.automobiles.aggreTT",
      "path": "aggrepropset"
    },
    {
      "__metadata": {
        "id": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/

```



```

measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-20T00%3A
00%3A00')",
    "uri": "https://analytics-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/
measurements(id='A2E9050BAFD7476B97AC4CCA0E309A35',time=datetime'2018-10-20T00%3A
00%3A00')",
    "type": "core.automobiles.aggrePST"
  },
  "id": "A2E9050BAFD7476B97AC4CCA0E309A35",
  "time": "/Date(1539993600000)/",
  "pressure": 20,
  "temperature": 34,
  "_ThingType": "core.automobiles:aggreTT",
  "_NPST": "aggrepropset",
  "thingType": "core.automobiles:aggreTT",
  "path": "aggrepropset"
}
]
}
}

```

## Related Information

[Measurements \[page 931\]](#)

# 17 Cloud-to-Cloud Interoperability

## Overview

SAP Leonardo IoT is enabled to interoperate with **Amazon IoT Core** and **Microsoft IoT Hub** for device connectivity, device management, and time series data ingestion. You can model and instantiate devices from these device connectivity providers as things. This enables, that an application, built on the Thing Model leverages ingested data for visualization, analysis, and related business processes. For more information, see [Cloud-to-Cloud Interoperability](#).

## Related Information

[Artifacts for Cloud-to-Cloud Interoperability \[page 942\]](#)

## 17.1 Artifacts for Cloud-to-Cloud Interoperability

After successfully completing the setup to enable cloud-to-cloud interoperability, you must now proceed with creation of artifacts for data ingestion.

### Base URI

- **Formal description:** `http://<server address>[:<port number>][<path>]`
- **Example for a base URI in a cloud foundry environment:** `https://sap-iotaexplore.iot-sap.cfapps.eu10.hana.ondemand.com/apiot-mds/`

## Methods

HTTP Method	Action	URI
<a href="#">POST</a>	<a href="#">Create Artifacts [page 943]</a>	/C2CArtifacts

## 17.1.1 Create Artifacts

With this service, you can create the technical artifacts that enable data ingestion for Cloud-to-Cloud interoperability. You can call this service when you onboard a tenant.

### Request

**URI:** `/C2CArtifacts`

**HTTP Method:** `POST`

**Permissions:** `<thing>.C2CAdmin.c`

### Request Parameters

None

### Request Example

`/C2CArtifacts`

### Payload

**Format:** `JSON`

In the request payload, you specify the tenant name for which you want to create the artifacts in the `<_tenant>` field.

```
{
    "_tenant" : "DEFAULT_TENANT"
}
```

### Response

**Format:** `JSON`

### Response Status and Error Codes

Code	Description
201	Technical artifacts created successfully

## Related Information

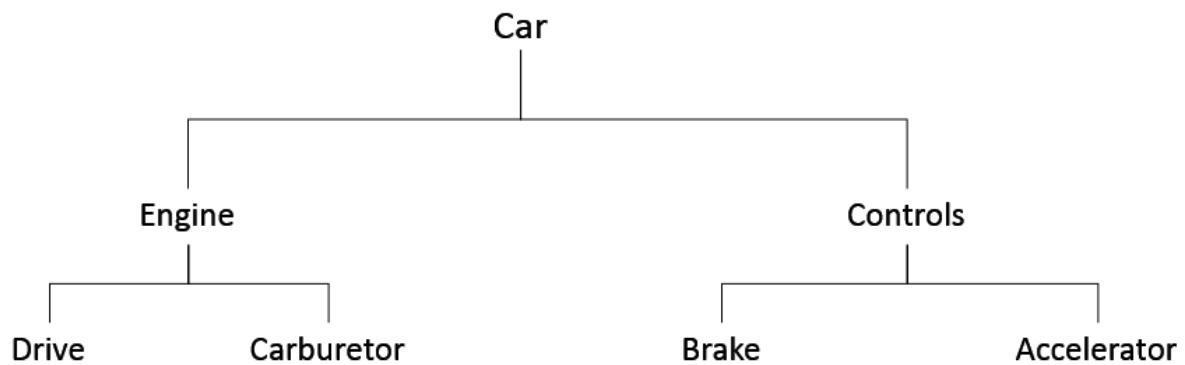
[Time Series Data from Device](#)

# 18 Thing Hierarchy

Hierarchy is a way of structuring and organizing objects in a tree structure, where every object except the top one (root) has exactly one immediate superior object and all objects may have one or multiple subordinate objects. This is called a parent-child relationship.

SAP Leonardo IoT services offer OData services to create, read, and maintain thing hierarchies. A Thing Hierarchy represents a group of things, with one dedicated thing at the top and potentially multiple levels of things below. It defines the parent-child relationship between these things.

The following diagram illustrates a hierarchical representation of a list of things:



## i Note

In the above example,

- **Drive** and **Carburetor** are child nodes of thing **Engine**.
- **Brake** and **Accelerator** are child nodes of thing **Controls**.
- **Engine** and **Controls** are child nodes of thing **Car**.

**Homogenous hierarchy:** The hierarchy contains the same object types. The only supported object type is Thing.

**Multiple hierarchies:** You can maintain multiple hierarchies for a specific tenant, and there exists only one root node in a hierarchy. A thing can belong to multiple hierarchies. However, a thing cannot appear multiple times in the same hierarchy.

**Hierarchy information:** Within a hierarchy you only create, update, or delete relationships between things. Thing instances themselves are not affected by operations performed on the hierarchy. For example, when you create a thing hierarchy, a parent-child relationship between the things is created. When you delete a thing within a hierarchy, only the parent-child relationship between the deleted thing and its associated things (if any) is removed. The thing instance is not deleted.

## Related Information

[Thing Hierarchy \[page 946\]](#)

# 18.1 Thing Hierarchy

Thing hierarchy related services

## Overview

The service allows you to create, read, and maintain hierarchy of related things.

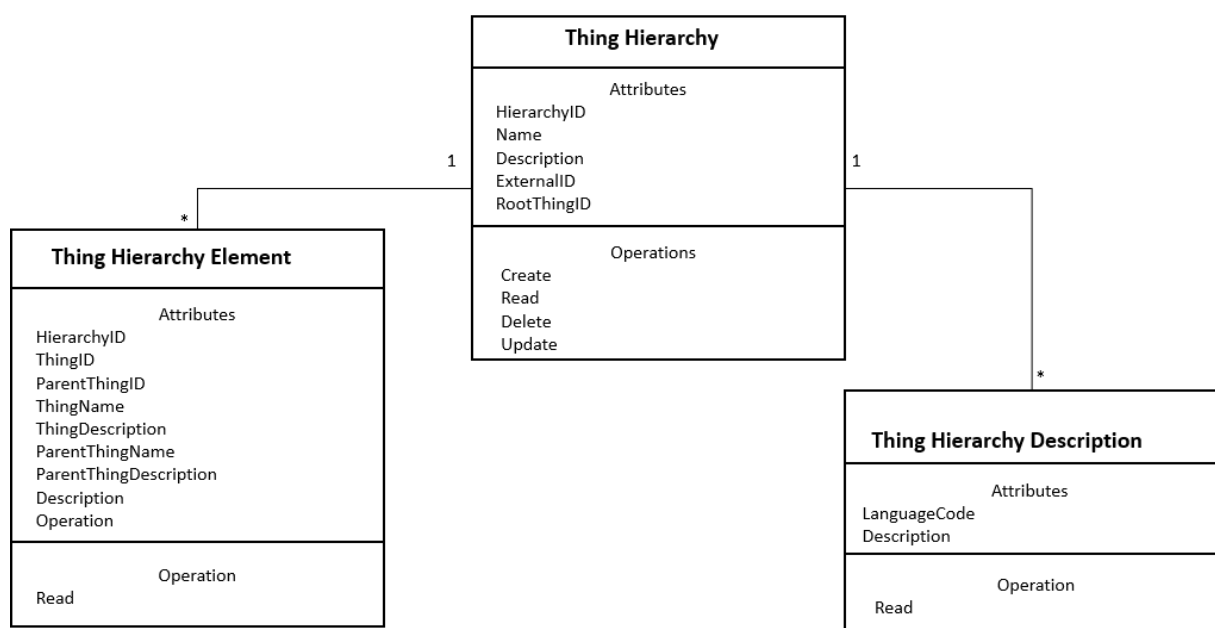
**OData Version: 2.0**

**Root URI:** `http://<server address>[:<port number>]/ThingHierarchy`

**Example for a Root URI in a cloud foundry environment:** `https://apiot-thing-hierarchy.cfapps.eu10.hana.ondemand.com/ThingHierarchy`

## Entity Data Model (EDM)

The following diagram shows the EDM of the Thing Hierarchy:



**Service Metadata URI:** `http://<server address>[:<port number>]/ThingHierarchy/v1/$metadata`

## Resources

Resource	Description	Path
<a href="#">Thing Hierarchy [page 947]</a>	Represents thing hierarchies	<code>/v1/ThingHierarchies</code>
<a href="#">Thing Hierarchy Element [page 973]</a>	Thing hierarchy elements	<code>/v1/ThingHierarchies?\$expand=ThingHierarchyElements</code>
<a href="#">Thing Hierarchy Description [page 979]</a>	Thing hierarchy descriptions	<code>/v1/ThingHierarchies?\$expand=ThingHierarchyDescriptions</code>

### 18.1.1 Thing Hierarchy

Represents thing hierarchies

**Resource Path:**`http://<server address>[:<port number>]/ThingHierarchy/v1/ThingHierarchies`

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scopes
<i>READ</i>	<a href="#">Read Metadata of Thing Hierarchy [page 950]</a>	<code>/ThingHierarchy/v1/\$metadata</code>	<code>&lt;ths&gt;.r</code>
<i>POST</i>	<a href="#">Create Thing Hierarchy [page 954]</a>	<code>/v1/ThingHierarchies</code>	<code>&lt;ths&gt;.c</code> <code>&lt;ohs&gt;.c</code>
<i>GET</i>	<a href="#">Read Thing Hierarchy [page 956]</a>	<code>/v1/ThingHierarchies ('&lt;thing hierarchy ID&gt;')</code>	<code>&lt;ths&gt;.r</code>
<i>READ</i>	<a href="#">Read Sub-hierarchy for a Thing [page 960]</a>	<code>/ThingHierarchy/v1/ThingHierarchies ('&lt;thing hierarchy ID&gt;')/ThingHierarchyElements?\$filter=ThingID eq '&lt;thing ID&gt;' &amp; \$inlinecount=allpages&amp; \$format=json&amp;sap.apptot.level=1</code>	<code>&lt;ths&gt;.r</code>

HTTP Method	Operation	URI	Scopes
<i>GET</i>	<a href="#">Read all Thing Hierarchies [page 965]</a>	/v1/ThingHierarchies	<ths>.r
<i>DELETE</i>	<a href="#">Delete Thing Hierarchy [page 967]</a>	/v1/ThingHierarchies ('<thing hierarchy ID>')	<ths>.d <ohs>.d
<i>UPDATE</i>	<a href="#">Update Thing Hierarchy [page 969]</a>	/v1/ThingHierarchies ('<thing hierarchy ID>')	<ths>.u <ohs>.u

## Thing Hierarchy Properties

Parameter	Data Type	Maximum Length	Description
HierarchyID	string	32	Unique identifier of the thing hierarchy generated by the system.
Name	string	255	Name of the thing hierarchy Note: <div data-bbox="1104 1102 1402 1482" data-label="Complex-Block"> <p><b>i Note</b></p> <p>Only the following characters are supported:</p> <ul style="list-style-type: none"> <li>Alphabets <b>a</b> through <b>z</b> and <b>A</b> through <b>Z</b></li> <li>Numeric digits <b>0</b> through <b>9</b></li> <li>Punctuation mark <b>underscore ( _ )</b></li> </ul> </div>
Description	string	255	Description of the thing hierarchy  The descriptive text provided in this field is ignored and is replaced with the descriptive text provided in the <b>Description</b> field of the <b>Thing Hierarchy Description</b> entity.



Parameter	Data Type	Maximum Length	Description
ExternalID	string	255	<p>Thing hierarchy identifier (which is not unique) from an external system. You can create multiple thing hierarchies with the same external ID.</p> <div> <p><b>i Note</b></p> <p>Only the following characters are supported:</p> <ul style="list-style-type: none"> <li>Alphabets <b>a</b> through <b>z</b> and <b>A</b> through <b>Z</b></li> <li>Numeric digits <b>0</b> through <b>9</b></li> <li>Punctuation marks <b>hyphen (-)</b>, <b>underscore (_)</b>, <b>colon (:)</b>, and <b>full-stop (.)</b></li> </ul> </div>
RootThingID	string	32	<p>Identifier of the thing in the root element of the thing hierarchy. The thing with parentThingID = 0 is considered as the root thing of the hierarchy. You cannot edit the root thing ID when you create or update a thing hierarchy.</p>

#### i Note

- If the description is maintained in the requested language, the service returns the payload with description.
- If the description is not maintained in the requested language, the service returns the payload with description maintained in default language (EN – English).
- If the description is not maintained in the default language (EN – English) as well, the service returns the payload with `null` value for the description.

## Related Information

[Thing Hierarchy Element \[page 973\]](#)

[Thing Hierarchy Description \[page 979\]](#)

## 18.1.1.1 Read Metadata of Thing Hierarchy

With this method, you can retrieve the thing hierarchy metadata.

### Request

**URI:** `/ThingHierarchy/v1/$metadata`

**Operation Type:** CRUD

**HTTP Method:** [\*GET\*](#)

**Permissions:** `<ths>.r`

### Request Parameters

None

### Request Example

`/ThingHierarchy/v1/$metadata`

### Response

#### Response Status and Error Codes

Code	Description
200	Thing Hierarchy metadata retrieved successfully.

### Payload

```
<EntityType Name="ThingHierarchyElement">
  <Documentation>
    <Summary>Details of thing hierarchy element entity type</Summary>
  </Documentation>
  <Key>
    <PropertyRef Name="HierarchyID"></PropertyRef>
    <PropertyRef Name="ThingID"></PropertyRef>
  </Key>
  <Property Name="HierarchyID" Type="Edm.String" Nullable="false"
    MaxLength="32">
    <Documentation>
      <Summary>Unique identifier of the thing hierarchy</Summary>
      <LongDescription>It is automatically generated by the
        system when creating a thing hierarchy.</LongDescription>
    </Documentation>
  </Property>
```

```

        <Property Name="ThingID" Type="Edm.String" Nullable="false"
MaxLength="255">
            <Documentation>
                <Summary>Unique identifier of the thing</Summary>
            </Documentation>
        </Property>
        <Property Name="ThingName" Type="Edm.String" MaxLength="255"
sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Name of the thing specified when creating a
thing</Summary>
                <LongDescription>It cannot be edited during create
(POST) or update (PATCH) of a thing hierarchy.</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingDescription" Type="Edm.String"
MaxLength="60" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Description of the thing specified when
creating a thing</Summary>
                <LongDescription>It cannot be edited during create
(POST) or update (PATCH) of a thing hierarchy.</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ParentThingID" Type="Edm.String" Nullable="true"
MaxLength="255">
            <Documentation>
                <Summary>Unique identifier of the parent thing</Summary>
            </Documentation>
        </Property>
        <Property Name="ParentThingName" Type="Edm.String"
MaxLength="255" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Name of the parent thing specified when
creating a thing</Summary>
                <LongDescription>It cannot be edited during create
(POST) or update (PATCH) of a thing hierarchy.</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ParentThingDescription" Type="Edm.String"
MaxLength="60" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Description of the parent thing specified when
creating a thing</Summary>
                <LongDescription>It cannot be edited during create
(POST) or update (PATCH) of a thing hierarchy.</LongDescription>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" MaxLength="255">
            <Documentation>
                <Summary>Description of the thing hierarchy element</
Summary>
            </Documentation>
        </Property>
        <Property Name="Operation" Type="Edm.String" MaxLength="6">
            <Documentation>
                <Summary>Operation to be performed on the thing
hierarchy element</Summary>
                <LongDescription>Indicates the add or delete operation
to be performed on the thing hierarchy element during PATCH request.</
LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingType" Type="Edm.String" Nullable="true"
MaxLength="81">
            <Documentation>
                <Summary>Thing type name</Summary>
            </Documentation>

```

```

        </Property>
        <Property Name="ThingTypeDescription" Type="Edm.String"
Nullable="true" MaxLength="60">
            <Documentation>
                <Summary>Thing type description</Summary>
            </Documentation>
        </Property>
        <NavigationProperty Name="ThingHierarchy"
Relationship="com.sap.apptot.ThingHierarchyElementToHierarchyAssociation"
FromRole="ThingHierarchyElement" ToRole="ThingHierarchy"></NavigationProperty>
    </EntityType>
    <EntityType Name="ThingHierarchy">
        <Documentation>
            <Summary>Details of thing hierarchy entity type</Summary>
        </Documentation>
        <Key>
            <PropertyRef Name="HierarchyID"></PropertyRef>
        </Key>
        <Property Name="HierarchyID" Type="Edm.String" Nullable="false"
MaxLength="32">
            <Documentation>
                <Summary>Unique identifier of the thing hierarchy</
Summary>
                <LongDescription>It is automatically generated by the
system when creating a thing hierarchy.</LongDescription>
            </Documentation>
        </Property>
        <Property Name="Name" Type="Edm.String" Nullable="true"
MaxLength="255">
            <Documentation>
                <Summary>Name of the thing hierarchy</Summary>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" Nullable="true"
MaxLength="255" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <Summary>Description of the thing hierarchy</Summary>
                <LongDescription>When creating a thing hierarchy, the
text provided in this field is ignored and replaced with the text provided in
the description field of the thing hierarchy description entity.</
LongDescription>
            </Documentation>
        </Property>
        <Property Name="ExternalID" Type="Edm.String" Nullable="true"
MaxLength="255">
            <Documentation>
                <Summary>Thing hierarchy identifier (which is not
unique) from an external system</Summary>
            </Documentation>
        </Property>
        <Property Name="RootThingID" Type="Edm.String" Nullable="true"
MaxLength="32">
            <Documentation>
                <Summary>Identifier of the thing in root element of the
thing hierarchy</Summary>
            </Documentation>
        </Property>
        <NavigationProperty Name="ThingHierarchyElements"
Relationship="com.sap.apptot.ThingHierarchyToElementAssociation"
FromRole="ThingHierarchy" ToRole="ThingHierarchyElement"></NavigationProperty>
        <NavigationProperty Name="ThingHierarchyDescriptions"
Relationship="com.sap.apptot.ThingHierarchyToDescriptionAssociation"
FromRole="ThingHierarchy" ToRole="Description"></NavigationProperty>
    </EntityType>
    <EntityType Name="Description">
        <Documentation>
            <Summary>Details of the thing hierarchy description entity
type</Summary>

```

```

        </Documentation>
        <Key>
            <PropertyRef Name="LanguageCode"></PropertyRef>
        </Key>
        <Property Name="LanguageCode" Type="Edm.String" Nullable="false"
MaxLength="2">
            <Documentation>
                <Summary>ISO code of the language of the thing hierarchy
description</Summary>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String" MaxLength="255">
            <Documentation>
                <Summary>Description of the thing hierarchy</Summary>
            </Documentation>
        </Property>
    </EntityType>
    <Association Name="ThingHierarchyToElementAssociation">
        <End Type="com.sap.appiot.ThingHierarchy" Multiplicity="1"
Role="ThingHierarchy"></End>
        <End Type="com.sap.appiot.ThingHierarchyElement"
Multiplicity="*" Role="ThingHierarchyElement"></End>
    </Association>
    <Association Name="ThingHierarchyElementToHierarchyAssociation">
        <End Type="com.sap.appiot.ThingHierarchyElement"
Multiplicity="*" Role="ThingHierarchyElement"></End>
        <End Type="com.sap.appiot.ThingHierarchy" Multiplicity="1"
Role="ThingHierarchy"></End>
    </Association>
    <Association Name="ThingHierarchyToDescriptionAssociation">
        <End Type="com.sap.appiot.ThingHierarchy" Multiplicity="1"
Role="ThingHierarchy"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
Role="Description"></End>
    </Association>
    <EntityContainer Name="ThingHierarchyContainer"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="ThingHierarchyElements"
EntityType="com.sap.appiot.ThingHierarchyElement">
            <Documentation>
                <Summary>Details of thing hierarchy elements entity set</
Summary>
            </Documentation>
        </EntitySet>
        <EntitySet Name="ThingHierarchies"
EntityType="com.sap.appiot.ThingHierarchy">
            <Documentation>
                <Summary>Details of thing hierarchies entity set</
Summary>
            </Documentation>
        </EntitySet>
        <EntitySet Name="Descriptions"
EntityType="com.sap.appiot.Description">
            <Documentation>
                <Summary>Details of the thing hierarchy descriptions
entity set</Summary>
            </Documentation>
        </EntitySet>
        <AssociationSet Name="ThingHierarchyToElementAssociation"
Association="com.sap.appiot.ThingHierarchyToElementAssociation">
            <End EntitySet="ThingHierarchies" Role="ThingHierarchy"></
End>
            <End EntitySet="ThingHierarchyElements"
Role="ThingHierarchyElement"></End>
        </AssociationSet>
        <AssociationSet
Name="ThingHierarchyElementToHierarchyAssociation"
Association="com.sap.appiot.ThingHierarchyElementToHierarchyAssociation">

```

```

        <End EntitySet="ThingHierarchyElements"
Role="ThingHierarchyElement"></End>
        <End EntitySet="ThingHierarchies" Role="ThingHierarchy"></
End>
        </AssociationSet>
        <AssociationSet Name="ThingHierarchyToDescriptionAssociation"
Association="com.sap.apiot.ThingHierarchyToDescriptionAssociation">
        <End EntitySet="ThingHierarchies" Role="ThingHierarchy"></
End>
        <End EntitySet="Descriptions" Role="Description"></End>
        </AssociationSet>
    </EntityContainer>

```

### 18.1.1.2 Create Thing Hierarchy

Creates a thing hierarchy

With this service, you can create thing hierarchy for a specific tenant. You can maintain multiple hierarchies for a specific tenant and there exists only one root node in a hierarchy. A thing can belong to multiple hierarchies. However, a thing cannot appear multiple times in the same hierarchy. The nodes within a hierarchy must be connected to at least one node and should not form cyclic reference. For example, if **Engine** and **Controls** are child nodes of the parent node **Car**, **Engine** cannot be the child node of **Controls**.

#### i Note

Using a single `POST` method, you can create a thing hierarchy with a maximum limit up to 20 levels and a maximum of 1000 objects. However, you can add more levels and objects to the thing hierarchy using the `PATCH` method.

When you create a thing hierarchy, you must provide the following details:

- Name of the hierarchy
- External ID
- Thing hierarchy elements
  - Thing ID
  - Parent thing ID
- Thing hierarchy description
  - ISO language code
  - Description

### Request

**URI:** `/v1/ThingHierarchies`

**Operation Type:** CRUD

**HTTP Method:** `POST`

**Permissions:**

`<ths>.c`

<ohs>.c

## Request Example

```
{
  "Name": "Car_XYZ_Hierarchy",
  "ExternalID": "CarXYZ-Hierarchy",
  "ThingHierarchyElements": [{
    "ThingID": "965B8AB45F624848BDEBD4837D093945",
    "ParentThingID": "0"
  }, {
    "ThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
    "ParentThingID": "965B8AB45F624848BDEBD4837D093945"
  }, {
    "ThingID": "88E2BA6E7512439E9A771EE626AB25A2",
    "ParentThingID": "965B8AB45F624848BDEBD4837D093945"
  }, {
    "ThingID": "43B87CB55B2D43A29AAB08AD885A3461",
    "ParentThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E"
  }, {
    "ThingID": "3467B81CFCE74A2B97E78000027D2105",
    "ParentThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E"
  }, {
    "ThingID": "800DEE78D35D43D0BB404BB27B98CDA8",
    "ParentThingID": "88E2BA6E7512439E9A771EE626AB25A2"
  }, {
    "ThingID": "4E60A688A13E43E28EC78A0606785435",
    "ParentThingID": "88E2BA6E7512439E9A771EE626AB25A2"
  }],
  "ThingHierarchyDescriptions": [{
    "LanguageCode": "en",
    "Description": "Desccribes hierarchy of units in the carXYZ"
  }]
}
```

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
201	Thing hierarchy created successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

```
{
  "d": {
    "__metadata": {
```

```

      "id": "https://appiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')",
      "uri": "https://appiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')",
      "type": "com.sap.appiot.ThingHierarchy"
    },
    "HierarchyID": "582E4688A7647F2CE200229BE0434C65"
  }
}

```

## Related Information

[Thing Hierarchy \[page 947\]](#)

[Thing Hierarchy Element \[page 973\]](#)

[Thing Hierarchy Description \[page 979\]](#)

### 18.1.1.3 Read Thing Hierarchy

Retrieves hierarchy details for a specific thing hierarchy

With this service, you can retrieve the hierarchy details for a specific thing hierarchy ID.

## Request

**URI:** /v1/ThingHierarchies('<thing hierarchy ID>')

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <ths>.r

## Request Parameters

Parameter	Required	Data Type	Description
HierarchyID	Yes	string	Unique thing hierarchy identifier

## Request Example

```
/ThingHierarchy/v1/ThingHierarchies('582E4688A7647F2CE200229BE0434C65')
```

Retrieves details of a specific thing hierarchy ID

```
/ThingHierarchy/v1/ThingHierarchies('582E4688A7647F2CE200229BE0434C65')?
$format=json&$expand=ThingHierarchyDescriptions,ThingHierarchyElements
```



Retrieves details of a specific thing hierarchy ID along with the thing details that are part of the thing hierarchy in [JSON](#) format

### Request Query Parameters

Parameter	Required	Data Type	Description
\$expand	No	string	Includes the related resources inline while retrieving the requested resources.  Supported associations of thing hierarchy that can be retrieved using \$expand are thing hierarchy elements and thing hierarchy description.
\$filter	No	String	Filter condition to be applied; all fields in the result set are valid.
\$format	No		Format to be applied on the result set.
\$select	No	String	Select condition for the data fields to be included in the result set; all fields in the result set are valid.

## Response

Format: [JSON](#)

### Response Status and Error Codes

Code	Description
200	Thing hierarchy retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure in the following examples:

#### Example 1

```
/ThingHierarchy/v1/ThingHierarchies('582E4688A7647F2CE200229BE0434C65')
```

```
{  
  "d": {
```

```

    "_metadata": {
      "id": "https://appiot-thing-hierarchy-
ci.cfapps.sap.hana.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')",
      "uri": "https://appiot-thing-hierarchy-
ci.cfapps.sap.hana.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')",
      "type": "com.sap.appiot.ThingHierarchy"
    },
    "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
    "Name": "Car_XYZ_Hierarchy",
    "Description": "Describes hierarchy of units in the carXYZ",
    "ExternalID": "CarXYZ-Hierarchy",
    "RootThingID": "965B8AB45F624848BDEBD4837D093945",
    "ThingHierarchyElements": {
      "_deferred": {
        "uri": "https://appiot-thing-hierarchy-
ci.cfapps.sap.hana.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')/ThingHierarchyElements"
      }
    },
    "ThingHierarchyDescriptions": {
      "_deferred": {
        "uri": "https://appiot-thing-hierarchy-
ci.cfapps.sap.hana.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')/ThingHierarchyDescriptions"
      }
    }
  }
}

```

## Example 2

/ThingHierarchy/v1/ThingHierarchies('582E4688A7647F2CE200229BE0434C65')?  
\$format=json&\$expand=ThingHierarchyDescriptions,ThingHierarchyElements

```

{
  "d": {
    "results": [
      {
        "_metadata": {
          "id": "https://appiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')",
          "uri": "https://appiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')",
          "type": "com.sap.appiot.ThingHierarchy"
        },
        "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
        "Name": "Car_XYZ_Hierarchy",
        "Description": "Describes hierarchy of units in the carXYZ",
        "ExternalID": "CarXYZ-Hierarchy",
        "RootThingID": "965B8AB45F624848BDEBD4837D093945",
        "ThingHierarchyElements": {
          "results": [
            {
              "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
              "ThingID": "965B8AB45F624848BDEBD4837D093945",
              "ThingName": "Car-XYZ",
              "ThingDescription": "A car of XYZ series",
              "ParentThingID": null,
              "ParentThingName": null,
              "ParentThingDescription": null,
              "Description": null,
              "Operation": null,
              "ThingType": "core.automobiles:ABC2Series",
            }
          ]
        }
      }
    ]
  }
}

```

```

    "ThingTypeDescription": "ABC2Series electric car"
  },
  {
    "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
    "ThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
    "ThingName": "Car-XYZ-Engine",
    "ThingDescription": "Engine of Car-XYZ",
    "ParentThingID": "965B8AB45F624848BDEBD4837D093945",
    "ParentThingName": "Car-XYZ",
    "ParentThingDescription": "A car of XYZ series",
    "Description": null,
    "Operation": null,
    "ThingType": "core.automobiles:ABC2Series",
    "ThingTypeDescription": "ABC2Series electric car"
  },
  {
    "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
    "ThingID": "88E2BA6E7512439E9A771EE626AB25A2",
    "ThingName": "Car-XYZ-ControlUnit",
    "ThingDescription": "ControlUnit of Car-XYZ",
    "ParentThingID": "965B8AB45F624848BDEBD4837D093945",
    "ParentThingName": "Car-XYZ",
    "ParentThingDescription": "A car of XYZ series",
    "Description": null,
    "Operation": null,
    "ThingType": "core.automobiles:ABC2Series",
    "ThingTypeDescription": "ABC2Series electric car"
  },
  {
    "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
    "ThingID": "43B87CB55B2D43A29AAB08AD885A3461",
    "ThingName": "Car-XYZ-DriveUnit",
    "ThingDescription": "DriveUnit of Car-XYZ",
    "ParentThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
    "ParentThingName": "Car-XYZ-Engine",
    "ParentThingDescription": "Engine of Car-XYZ",
    "Description": null,
    "Operation": null,
    "ThingType": "core.automobiles:ABC2Series",
    "ThingTypeDescription": "ABC2Series electric car"
  },
  {
    "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
    "ThingID": "3467B81CFCE74A2B97E78000027D2105",
    "ThingName": "Car-XYZ-Carburetor",
    "ThingDescription": "Carburetor of Car-XYZ",
    "ParentThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
    "ParentThingName": "Car-XYZ-Engine",
    "ParentThingDescription": "Engine of Car-XYZ",
    "Description": null,
    "Operation": null,
    "ThingType": "core.automobiles:ABC2Series",
    "ThingTypeDescription": "ABC2Series electric car"
  },
  {
    "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
    "ThingID": "800DEE78D35D43D0BB404BB27B98CDA8",
    "ThingName": "Car-XYZ-Brake",
    "ThingDescription": "Brake of Car-XYZ",
    "ParentThingID": "88E2BA6E7512439E9A771EE626AB25A2",
    "ParentThingName": "Car-XYZ-ControlUnit",
    "ParentThingDescription": "ControlUnit of Car-XYZ",
    "Description": null,
    "Operation": null,
    "ThingType": "core.automobiles:ABC2Series",
    "ThingTypeDescription": "ABC2Series electric car"
  },
  {

```

```

    "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
    "ThingID": "4E60A688A13E43E28EC78A0606785435",
    "ThingName": "Car-XYZ-Accelerator",
    "ThingDescription": "Accelerator of Car-XYZ",
    "ParentThingID": "88E2BA6E7512439E9A771EE626AB25A2",
    "ParentThingName": "Car-XYZ-ControlUnit",
    "ParentThingDescription": "ControlUnit of Car-XYZ",
    "Description": null,
    "Operation": null,
    "ThingType": "core.automobiles:ABC2Series",
    "ThingTypeDescription": "ABC2Series electric car"
  }
]
},
"ThingHierarchyDescriptions": {
  "results": [
    {
      "LanguageCode": "en",
      "Description": "Describes hierarchy of units in the carXYZ"
    }
  ]
}
}
}
}
```

## Related Information

Thing Hierarchy [page 947]

Thing Hierarchy Element [page 973]

## Thing Hierarchy Description [page 979]

#### 18.1.1.4 Read Sub-hierarchy for a Thing

With this method, you can retrieve the sub-hierarchy for the specified thing ID in a hierarchy.

The following new custom query parameters are introduced to retrieve the sub-hierarchy:

- `sap.appiot.level`: The value of this parameter indicates the number of levels up to which child nodes or parent nodes of the specified thing ID are retrieved. Supported values are 0 and any positive integer. If you do not specify this parameter in the request URL or you assign a value 0 for this parameter in the request URL, the method retrieves all child nodes of the specified thing ID.
- `sap.appiot.order`: The value of this parameter indicates whether child nodes or parent nodes of the specified thing ID are retrieved. Supported values are **ascending** and **descending**. If you do not specify this parameter in the request URL, the method retrieves all child nodes of the specified thing ID up to the level mentioned in the parameter `sap.appiot.level`.

For example, consider the following thing hierarchy:



If `sap.apptot.level = 2`, `sap.apptot.order = descending` and, thing ID = `72280984D25430CBDD6638163061A03`, the method will fetch a sub-hierarchy with child nodes `620CD96C317B46AD9027E39DEAD3D3` and `56A32305301A4BFB96ED5D5F736EC19F`.

If `sap.apptot.level = 1`, `sap.apptot.order = ascending` and, thing ID = `72280984D25430CBDD6638163061A03`, the method will fetch a sub-hierarchy with parent node `D666B32455A48AE8829C42765A3141`.

#### Note

- The query parameter `$filter` is mandatory in the request URL and the only supported field for `$filter` is thing ID.
- The custom query parameters are not supported as a valid field in query request parameter `$filter`, `$orderby`, and `$select`.

## Request

**URI:** `/ThingHierarchy/v1/ThingHierarchies('<thing hierarchy ID>')/ThingHierarchyElements?$filter=ThingID eq '<thing ID>' &$inlinecount=allpages&$format=json&sap.apptot.level=1`

**Operation Type:** CRUD

HTTP Method: [GET](#)

Permissions: <ths>.r

## Request Parameters

Parameter	Required	Data Type	Description
thing hierarachy ID	Yes	String	Unique thing hierarchy identifier
thing ID	Yes	String	Unique thing identifier
sap.apptot.level	No	Integer	Custom query parameter and is not available in the metadata of thing hierarchy.
sap.apptot.order	No	String	Custom query parameter and is not available in the metadata of thing hierarchy.

## Query Request Parameters

Parameter	Required	Data Type	Description
\$filter	No	String	Filter condition to be applied; the only field supported is thing ID.
\$inlinecount	No	String	Total number of hierarchies available; permissible values are <a href="#">allpages</a> and <a href="#">none</a> . The following results apply if this parameter is used in conjunction with the other query parameters: <ul style="list-style-type: none"><li>If you use this with the \$top and \$skip parameters, the result contains the total count of hierarchies available.</li><li>If you use this with \$filter parameter, the result contains the total count of hierarchies based on the defined filter condition.</li></ul>
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; all fields in the result set are valid. To sort the result set, use <b>asc</b> for the ascending order and <b>desc</b> for the descending order.
\$skip	No	Integer	Number of records to exclude from the result set.
\$top	No	Integer	Number of records to include in the result set.
\$select	No	String	Select condition for the data fields to be included in the result set; all fields in the result set are valid.
\$skiptoken	No	Integer	Identifies a starting point in the collection of entities identified by the URI containing the \$skiptoken parameter.
\$format	No	String	Format to be applied on the result set. For example, <a href="#">json</a> or <a href="#">xml</a> .

## Request Example

```
/ThingHierarchy/v1/ThingHierarchies('0C714859AFC2E120E10000000A619C52')/  
ThingHierarchyElements?$filter=ThingID eq '720280984D25430CBDD6638163061A03'&  
$inlinecount=allpages&$format=json&sap.apptot.level=1
```

## Response

### Response Status and Error Codes

Code	Description
200	[Description]
501	Not implemented (other fields are not supported or implemented for \$filter query parameter)

## Payload

```
/ThingHierarchy/v1/ThingHierarchies('0C714859AFC2E120E10000000A619C52')/  
ThingHierarchyElements?$filter=ThingID eq '720280984D25430CBDD6638163061A03' &  
$inlinecount=allpages&$format=json&sap.apptot.level=1
```

Retrieves the sub-hierarchy of thing ID 720280984D25430CBDD6638163061A03 in the hierarchy 0C714859AFC2E120E10000000A619C52. The number of levels retrieved is 1.

```
{  
  "d": {  
    "__count": "1",  
    "results": [{  
      "__metadata": {  
        "id": "http://localhost:8080/ThingHierarchy/v1/  
ThingHierarchyElements(HierarchyID='0C714859AFC2E120E10000000A619C52',ThingID='62  
0CD96C317B46AD9027E39DEAD203D3')",  
        "uri": "http://localhost:8080/ThingHierarchy/v1/  
ThingHierarchyElements(HierarchyID='0C714859AFC2E120E10000000A619C52',ThingID='62  
0CD96C317B46AD9027E39DEAD203D3')",  
        "type": "com.sap.apptot.ThingHierarchyElement"  
      },  
      "HierarchyID": "0C714859AFC2E120E10000000A619C52",  
      "ThingID": "620CD96C317B46AD9027E39DEAD203D3",  
      "ThingName": "puttest",  
      "ThingDescription": "English",  
      "ParentThingID": "720280984D25430CBDD6638163061A03",  
      "ParentThingName": "Sybase.ASE.Thing0312222016_154132_478_1_-.23.",  
      "ParentThingDescription": "Thing for analysis on Sybase-ASE",  
      "Description": "HandsonHierarchy desc",  
      "Operation": null,  
      "ThingType": "core.automobiles:ABC2Series",  
      "ThingTypeDescription": "ABC2Series electric car"  
    },  
    "ThingHierarchy": {  
      "__deferred": {  
        "uri": "http://localhost:8080/ThingHierarchy/v1/  
ThingHierarchyElements(HierarchyID='0C714859AFC2E120E10000000A619C52',ThingID='62  
0CD96C317B46AD9027E39DEAD203D3')/ThingHierarchy"  
      }  
    }  
  }  
}]  
}
```

```
/ThingHierarchy/v1/ThingHierarchies('0C714859AFC2E120E10000000A619C52')/  
ThingHierarchyElements?$filter=ThingID eq '56A32305301A4BFB96ED5D5F736EC19F' &
```

```
$format=json&$inlinecount=allpages&
$expand=ThingHierarchy&sap.apptot.order=ascending&sap.apptot.level=0
```

```
{
  "d": {
    "__count": "3",
    "results": [
      {
        "__metadata": {
          "id": "http://localhost:8092/ThingHierarchy/v1/
ThingHierarchyElements(HierarchyID='0C714859AFC2E120E10000000A619C52',ThingID='56
A32305301A4BFB96ED5D5F736EC19F')",
          "uri": "http://localhost:8092/ThingHierarchy/v1/
ThingHierarchyElements(HierarchyID='0C714859AFC2E120E10000000A619C52',ThingID='56
A32305301A4BFB96ED5D5F736EC19F')",
          "type": "com.sap.apptot.ThingHierarchyElement"
        },
        "HierarchyID": "0C714859AFC2E120E10000000A619C52",
        "ThingID": "56A32305301A4BFB96ED5D5F736EC19F",
        "ThingName": "SESeriesA002",
        "ThingDescription": "English",
        "ParentThingID": "620CD96C317B46AD9027E39DEAD203D3",
        "ParentThingName": "puttest",
        "ParentThingDescription": "English",
        "Description": "HandsonHierarchy desc",
        "Operation": null,
        "ThingType": "com.sap.local1.iot.tml:ABC2Series",
        "ThingTypeDescription": "English",
        "ThingHierarchy": {
          "HierarchyID": "0C714859AFC2E120E10000000A619C52",
          "Name": "HandsonHierarchy",
          "Description": "HandsonHierarchy desc",
          "ExternalID": "ExternalID",
          "RootThingID": "D6606B32455A48AE8829C42765A03141"
        }
      },
      {
        "__metadata": {
          "id": "http://localhost:8092/ThingHierarchy/v1/
ThingHierarchyElements(HierarchyID='0C714859AFC2E120E10000000A619C52',ThingID='62
0CD96C317B46AD9027E39DEAD203D3')",
          "uri": "http://localhost:8092/ThingHierarchy/v1/
ThingHierarchyElements(HierarchyID='0C714859AFC2E120E10000000A619C52',ThingID='62
0CD96C317B46AD9027E39DEAD203D3')",
          "type": "com.sap.apptot.ThingHierarchyElement"
        },
        "HierarchyID": "0C714859AFC2E120E10000000A619C52",
        "ThingID": "620CD96C317B46AD9027E39DEAD203D3",
        "ThingName": "puttest",
        "ThingDescription": "English",
        "ParentThingID": "720280984D25430CBDD6638163061A03",
        "ParentThingName": "Sybase.ASE.Thing0312222016_154132_478_1_-:
23.",
        "ParentThingDescription": "Thing for analysis on Sybase-ASE",
        "Description": "HandsonHierarchy desc",
        "Operation": null,
        "ThingType": "com.sap.timeputt:TT1",
        "ThingTypeDescription": "English",
        "ThingHierarchy": {
          "HierarchyID": "0C714859AFC2E120E10000000A619C52",
          "Name": "HandsonHierarchy",
          "Description": "HandsonHierarchy desc",
          "ExternalID": "ExternalID",
          "RootThingID": "D6606B32455A48AE8829C42765A03141"
        }
      }
    ]
  },
  {

```



```

      "__metadata": {
        "id": "http://localhost:8092/ThingHierarchy/v1/ThingHierarchyElements(HierarchyID='0C714859AFC2E120E10000000A619C52',ThingID='720280984D25430CBDD6638163061A03')",
        "uri": "http://localhost:8092/ThingHierarchy/v1/ThingHierarchyElements(HierarchyID='0C714859AFC2E120E10000000A619C52',ThingID='720280984D25430CBDD6638163061A03')",
        "type": "com.sap.apptot.ThingHierarchyElement"
      },
      "HierarchyID": "0C714859AFC2E120E10000000A619C52",
      "ThingID": "720280984D25430CBDD6638163061A03",
      "ThingName": "Sybase.ASE.Thing0312222016_154132_478_1_-:23.",
      "ThingDescription": "Thing for analysis on Sybase-ASE",
      "ParentThingID": "D6606B32455A48AE8829C42765A03141",
      "ParentThingName": "Sybase.ASE.Thing0312222016_154132_478_1_-:.",
      "ParentThingDescription": "Thing for analysis on Sybase-ASE",
      "Description": "HandsonHierarchy desc",
      "Operation": null,
      "ThingType": "sybase.test.alldatatype:ThingType",
      "ThingTypeDescription": "Thing for analysis on Sybase-ASE",
      "ThingHierarchy": {
        "HierarchyID": "0C714859AFC2E120E10000000A619C52",
        "Name": "HandsonHierarchy",
        "Description": "HandsonHierarchy desc",
        "ExternalID": "ExternalID",
        "RootThingID": "D6606B32455A48AE8829C42765A03141"
      }
    }
  ]
}

```

## Related Information

[Thing Hierarchy \[page 947\]](#)

[Thing Hierarchy Element \[page 973\]](#)

[Thing Hierarchy Description \[page 979\]](#)

### 18.1.1.5 Read all Thing Hierarchies

Retrieves all thing hierarchies

With this service, you can retrieve all the thing hierarchies for a specific tenant.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/v1/ThingHierarchies`

Operation Type: CRUD

HTTP Method: [GET](#)

Permissions: <ths>.r

## Request Query Parameters

Parameter	Required	Data Type	Description
\$filter	No	String	Filter condition to be applied; all fields in the result set are valid.
\$inlinecount	No	String	Total number of hierarchies available; permissible values are <a href="#">allpages</a> and <a href="#">none</a> . The following results apply if this parameter is used in conjunction with the other query parameters: <ul style="list-style-type: none"><li>If you use this with the \$top and \$skip parameters, the result contains the total count of hierarchies available.</li><li>If you use this with \$filter parameter, the result contains the total count of hierarchies based on the defined filter condition.</li></ul>
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; all fields in the result set are valid. To sort the result set, use <b>asc</b> for the ascending order and <b>desc</b> for the descending order.
\$skip	No	Integer	Number of records to exclude from the result set.
\$top	No	Integer	Number of records to include in the result set.
\$select	No	String	Select condition for the data fields to be included in the result set; all fields in the result set are valid.
\$skiptoken	No	Integer	Identifies a starting point in the collection of entities identified by the URI containing the \$skiptoken parameter.
\$format	No	String	Format to be applied on the result set. For example, <a href="#">json</a> or <a href="#">xml</a> .

## Request Example

```
/ThingHierarchy/v1/ThingHierarchies?$format=json
```

## Response

Format: [JSON](#)

## Response Status and Error Codes

Code	Description
200	Thing hierarchies retrieved successfully.

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

```
{
  "d": {
    "results": [{
      "metadata": {
        "id": "https://appiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')",
        "uri": "https://appiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')",
        "type": "com.sap.appiot.ThingHierarchy"
      },
      "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
      "Name": "Car_XYZ_Hierarchy",
      "Description": "Desccribes hierarchy of units in the carXYZ",
      "ExternalID": "CarXYZ-Hierarchy",
      "RootThingID": "965B8AB45F624848BDEBD4837D093945",
      "ThingHierarchyElements": {
        "deferred": {
          "uri": "https://appiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')/ThingHierarchyElements"
        }
      },
      "ThingHierarchyDescriptions": {
        "deferred": {
          "uri": "https://appiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchies('582E4688A7647F2CE200229BE0434C65')/ThingHierarchyDescriptions"
        }
      }
    }]
  }
}
```

## Related Information

[Thing Hierarchy \[page 947\]](#)

[Thing Hierarchy Element \[page 973\]](#)

[Thing Hierarchy Description \[page 979\]](#)

### 18.1.1.6 Delete Thing Hierarchy

Deletes thing hierarchy with a specific ID

The ETag value is not mandatory to delete a thing hierarchy. However, to delete a thing hierarchy using ETag, you must retrieve the ETag value of the thing hierarchy using the GET (/ThingHierarchy/v1/ThingHierarchies('<thing hierarchy ID>')) service. The ETag value is assigned to the request

header parameter `If-Match` in the `DELETE` request. If the `ETag` value is not correct, the server sends HTTP response code 412.

## Request

**URI:** `/v1/ThingHierarchies('<thing hierarchy ID>')`

**Operation Type:** CRUD

**HTTP Method:** `DELETE`

**Permissions:**

`<ths>.d`

`<ohs>.d`

### Request Header Parameters

Parameter	Required	Values
<code>Accept-Language</code>	No	Language of the thing type description. The default language is en.
<code>If-Match</code>	No	Latest <code>ETag</code> value of a specific thing hierarchy that is updated using the <code>PATCH</code> request.

### Request Parameters

Parameter	Required	Data Type	Description
<code>HierarchyID</code>	Yes	string	Unique thing hierarchy identifier

### Request Example

```
/ThingHierarchy/v1/ThingHierarchies('582E4688A7647F2CE200229BE0434C65')
```

## Response

**Format:** `JSON`

### Response Status and Error Codes

Code	Description
204	Thing hierarchy deleted successfully.

## Related Information

[Thing Hierarchy \[page 947\]](#)

[Thing Hierarchy Element \[page 973\]](#)

[Thing Hierarchy Description \[page 979\]](#)

### 18.1.1.7 Update Thing Hierarchy

Updates thing hierarchy with a specific ID

You can update the thing hierarchy name, external ID, and the language dependent descriptions using this service for a particular thing hierarchy. In addition, you can add new nodes or delete existing nodes in the existing thing hierarchy. The new nodes cannot appear multiple times in the same hierarchy and the nodes within a hierarchy must be connected to at least one node and should not form cyclic reference. The supported operations in this service are *ADD* and *DELETE*. The system does not allow you to delete objects with child nodes, you can only delete the child nodes. The system first executes all *DELETE* operations and then executes *ADD* operations. The sequence of operations you specify in the request payload does not alter the sequence of operations performed at the backend. You can perform all *PATCH* operations only on the existing state of the hierarchy.

The ETag value is not mandatory to update a thing hierarchy. However, to update a thing hierarchy using ETag, you must retrieve the ETag value of the thing hierarchy using the GET (/ThingHierarchy/v1/ThingHierarchies('<thing hierarchy ID>')) service. The ETag value is assigned to the request header parameter If-Match in the PATCH request. If the ETag value is not correct, the server sends HTTP response code 412 .

## Request

**URI:** /v1/ThingHierarchies('<thing hierarchy ID>')

**Operation Type:** CRUD

**HTTP Method:** *PATCH*

**Permissions:**

<ths>.u

<ohs>.u

## Request Header Parameters

Parameter	Required	Values
Accept-Language	No	Language of the thing type description. The default language is en.

Parameter	Required	Values
If-Match	No	Latest ETag value of a specific thing hierarchy that is updated using the PATCH request.

## Request Parameters

Parameter	Required	Data Type	Description
HierarchyID	Yes	string	Unique thing hierarchy identifier

## Request Example

### Example 1

/ThingHierarchy/v1/ThingHierarchies('582E4688A7647F2CE200229BE0434C65')

```
{
  "Name": "Car XYZ_Hierarchy",
  "ExternalID": "CarXYZ-Hierarchy",
  "ThingHierarchyElements": [{
    "ThingID": "3467B81CFCE74A2B97E78000027D2105",
    "ParentThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
    "Operation": "DELETE"
  }, {
    "ThingID": "4E60A688A13E43E28EC78A0606785435",
    "ParentThingID": "88E2BA6E7512439E9A771EE626AB25A2",
    "Operation": "DELETE"
  }],
  "ThingHierarchyDescriptions": [{
    "LanguageCode": "en",
    "Description": "Desccribes hierarchy of units in the updated carXYZ"
  }, {
    "LanguageCode": "de",
    "Description": "Beschreibung der Hierarchie carXYZ"
  }]
}
```

In this example, the name, external ID, and the description in language code EN are modified for the thing hierarchy. In addition, two of the nodes are deleted from a different parent.

### Example 2

/ThingHierarchy/v1/ThingHierarchies('582E4688A7647F2CE200229BE0434C65')

```
{
  "ThingHierarchyElements": [{
    "ThingID": "3467B81CFCE74A2B97E78000027D2105",
    "ParentThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
    "Operation": "ADD"
  }, {
    "ThingID": "800DEE78D35D43D0BB404BB27B98CDA8",
    "ParentThingID": "88E2BA6E7512439E9A771EE626AB25A2",
    "Operation": "DELETE"
  }]
}
```

In this example, a new node is added and an existing node is deleted.

### Example 3

/ThingHierarchy/v1/ThingHierarchies('582E4688A7647F2CE200229BE0434C65')

```
{
  "ThingHierarchyElements": [{
    "ThingID": "90589958BF6D530DE20017D91D3A7FDC",
    "ParentThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
    "Operation": "ADD"
  }, {
    "ThingID": "90589958BF6D530DE20017D91D3A7FDC",
    "ParentThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
    "Operation": "DELETE"
  }]
}
```

In this example, operations are not performed on the existing state of the hierarchy, as a new node that is not existing in the hierarchy is added and the same is deleted. Hence, the response code for this request is 400, as the object that does not exist is specified for deletion.

#### Example 4

/ThingHierarchies('C333A558EF82AC01E200D797B4576E03')

```
{
  "ThingHierarchyElements": [
    {
      "ThingID": "88E2BA6E7512439E9A771EE626AB25A2",
      "ParentThingID": "965B8AB45F624848BDEBD4837D093945",
      "Operation": "DELETE"
    },
    {
      "ThingID": "800DEE78D35D43D0BB404BB27B98CDA8",
      "ParentThingID": "88E2BA6E7512439E9A771EE626AB25A2",
      "Operation": "DELETE"
    },
    {
      "ThingID": "4E60A688A13E43E28EC78A0606785435",
      "ParentThingID": "88E2BA6E7512439E9A771EE626AB25A2",
      "Operation": "DELETE"
    }
  ]
}
```

In this example, the thing 88E2BA6E7512439E9A771EE626AB25A2 is specified for deletion. However, the thing contains two child nodes. Only after deleting the child nodes, the system allows you to delete the parent thing. Hence, in the payload of the request, you can also mention the child nodes (800DEE78D35D43D0BB404BB27B98CDA8 and 4E60A688A13E43E28EC78A0606785435) to be deleted.

#### Example 5

/ThingHierarchies('16ACA6581B8C530DE20017D91D3A7FDC')

```
{
  "ThingHierarchyElements": [
    {
      "ThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
      "ParentThingID": "965B8AB45F624848BDEBD4837D093945",
      "Operation": "DELETE"
    },
    {
      "ThingID": "88E2BA6E7512439E9A771EE626AB25A2",
      "ParentThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
      "Operation": "DELETE"
    }
  ]
}
```

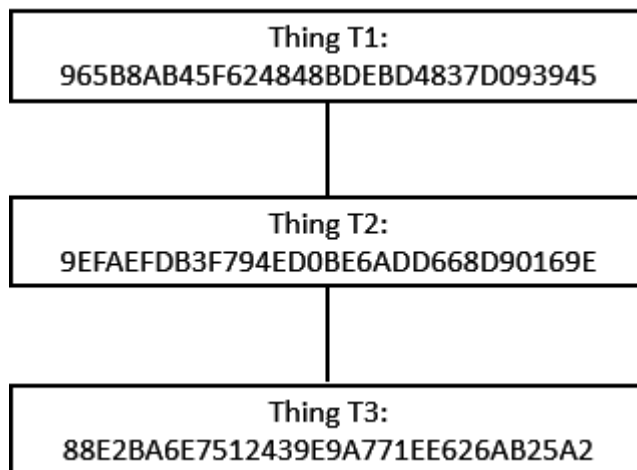
```

    "ThingID": "43B87CB55B2D43A29AAB08AD885A3461",
    "ParentThingID": "965B8AB45F624848BDEBD4837D093945",
    "Operation": "ADD"
  },
  {
    "ThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
    "ParentThingID": "43B87CB55B2D43A29AAB08AD885A3461",
    "Operation": "ADD"
  },
  {
    "ThingID": "88E2BA6E7512439E9A771EE626AB25A2",
    "ParentThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
    "Operation": "ADD"
  }
]
}

```

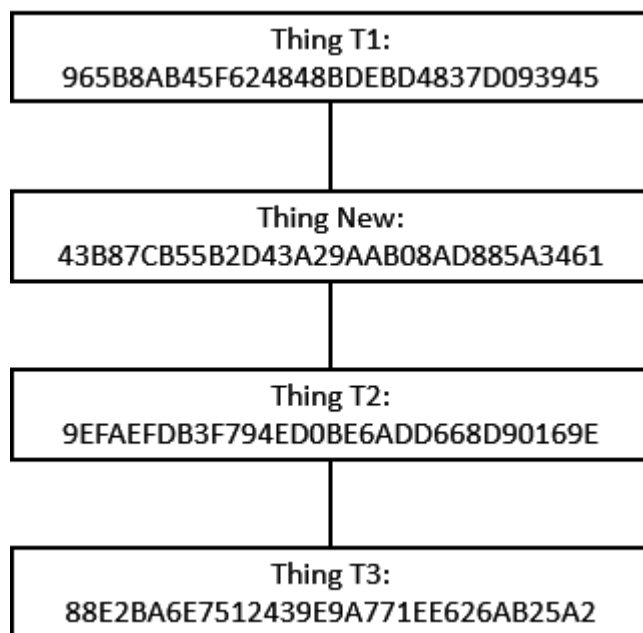
In this example, a new node (thing ID 43B87CB55B2D43A29AAB08AD885A3461) is added as an intermediate node.

The following diagram illustrates the existing hierarchy:





You might want to add an intermediate node and update the hierarchy as depicted in the following diagram:



Therefore, in the payload of the request, you specify the following:

1. Delete the child nodes **Thing T2** and **Thing T3**
2. Add the new intermediate node **Thing New** to the parent node **Thing T1**
3. Add the child node **Thing T2** to its new parent node **Thing New**
4. Add the child node **Thing T3** to its parent node **Thing T2**

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
204	Thing hierarchy updated successfully.

## 18.1.2 Thing Hierarchy Element

Represents one or more elements of an individual thing hierarchy

**Resource Path:**`http://<server address>[:<port number>]/ThingHierarchy/v1/ThingHierarchyElements`

## Operations

### CRUD Operations

HTTP Method	Operation	URI
GET	<a href="#">Read Thing Hierarchy Elements [page 975]</a>	/v1/ThingHierarchyElements

### Thing Hierarchy Element Properties

Parameter	Data Type	Maximum Length	Description
HierarchyID	String	32	Unique identifier of the thing hierarchy generated by the system
ThingID	String	255	Unique identifier of the thing generated by the system during creation of a thing.
ThingName	String	255	Name of the thing specified when you create a thing. You cannot edit the name when you create or update the thing hierarchy.
ThingDescription	String	255	Description of the thing specified when you create a thing. You cannot edit the description when you create or update the thing hierarchy.
ParentThingID	String	255	Unique identifier of the parent thing generated by the system during creation of a thing.
ParentThingName	String	255	Name of the parent thing specified when you create a thing. You cannot edit the name when you create or update the thing hierarchy.
ParentThingDescription	String	60	Description of the parent thing. You cannot edit the description when you create or update the thing hierarchy.
Description	String	255	Description of the thing hierarchy element

Parameter	Data Type	Maximum Length	Description
Operation	String	6	Operation to be performed on the thing hierarchy element. The supported operations are 'add' and 'delete' performed on the thing hierarchy element during PATCH request.

## Related Information

[Thing Hierarchy \[page 947\]](#)

[Thing Hierarchy Description \[page 979\]](#)

### 18.1.2.1 Read Thing Hierarchy Elements

Retrieves thing hierarchy elements for a specific thing ID

With this service, you can view all hierarchies to which a specific thing is associated. To retrieve the information, you must use `$filter` on thing ID and `$expand` on thing hierarchy ID. In addition, you can also retrieve the immediate child nodes of a thing in a specific hierarchy using `$filter` on parent thing ID and thing hierarchy ID.

## Request

**URI:** `/v1/ThingHierarchyElements?$filter=ThingID eq ('<thing ID>')&$expand=ThingHierarchy&$format=json`

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** `<ths>.r`

## Request Parameters

Parameter	Required	Data Type	Description
thingID	Yes	string	Unique thing identifier

## Request Example

```
/ThingHierarchy/v1/ThingHierarchyElements?$filter=ThingID eq
'9EFAEFDB3F794ED0BE6ADD668D90169E'&$expand=ThingHierarchy&$format=json
```

Retrieves all hierarchies for a specific thing

```
/ThingHierarchy/v1/ThingHierarchyElements?$filter=ParentThingID eq  
'965B8AB45F624848BDEBD4837D093945' and HierarchyID eq  
'582E459DA7647F2CE200229BE0434C65'&$format=json
```

Retrieves immediate child nodes for a thing in a specific thing hierarchy

### Request Query Parameters

Parameter	Required	Data Type	Description
\$expand	Yes	string	Includes the related resources inline while retrieving the requested resources.
\$filter	Yes	string	Filter condition to be applied; all fields in the result set are valid. It is mandatory to specify thing ID or parent thing ID in the filter condition. If you specify thing ID, \$expand is mandatory.
\$format	No		Format to be applied on the result set.

### Response

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

### Response Status and Error Codes

Code	Description
200	Thing hierarchy retrieved successfully.

### Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

### All hierarchies for a thing

Request:

```
/ThingHierarchy/v1/ThingHierarchyElements?$filter=ThingID eq  
'9EFAEFDB3F794ED0BE6ADD668D90169E'&$expand=ThingHierarchy&$format=json
```

Response:

```
{
```

```

"d": {
  "results": [
    {
      "_metadata": {
        "id": "https://apiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchyElements(HierarchyID='582E459DA7647F2CE200229BE0434C65',ThingID='9E
FAEFDB3F794ED0BE6ADD668D90169E')",
        "uri": "https://apiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchyElements(HierarchyID='582E459DA7647F2CE200229BE0434C65',ThingID='9E
FAEFDB3F794ED0BE6ADD668D90169E')",
        "type": "com.sap.apiot.ThingHierarchyElement"
      },
      "HierarchyID": "582E459DA7647F2CE200229BE0434C65",
      "ThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
      "ThingName": "Car-XYZ-Engine",
      "ThingDescription": "Engine of Car-XYZ",
      "ParentThingID": "965B8AB45F624848BDEBD4837D093945",
      "ParentThingName": "Car-XYZ",
      "ParentThingDescription": "A car of XYZ series",
      "Description": "Desccribes hierarchy of units in the carXYZ",
      "Operation": null,
      "ThingType": "core.automobiles:ABC2Series",
      "ThingTypeDescription": "ABC2Series electric car",
      "ThingHierarchy": {
        "HierarchyID": "582E459DA7647F2CE200229BE0434C65",
        "Name": "Car_XYZ_Hierarchy",
        "Description": "Desccribes hierarchy of units in the carXYZ",
        "ExternalID": "CarXYZ-Hierarchy",
        "RootThingID": "965B8AB45F624848BDEBD4837D093945"
      }
    },
    {
      "_metadata": {
        "id": "https://apiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchyElements(HierarchyID='582E4688A7647F2CE200229BE0434C65',ThingID='9E
FAEFDB3F794ED0BE6ADD668D90169E')",
        "uri": "https://apiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchyElements(HierarchyID='582E4688A7647F2CE200229BE0434C65',ThingID='9E
FAEFDB3F794ED0BE6ADD668D90169E')",
        "type": "com.sap.apiot.ThingHierarchyElement"
      },
      "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
      "ThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
      "ThingName": "Car-XYZ-Engine",
      "ThingDescription": "Engine of Car-XYZ",
      "ParentThingID": "965B8AB45F624848BDEBD4837D093945",
      "ParentThingName": "Car-XYZ",
      "ParentThingDescription": "A car of XYZ series",
      "Description": "Desccribes hierarchy of units in the carXYZ",
      "Operation": null,
      "ThingType": "core.automobiles:ABC2Series",
      "ThingTypeDescription": "ABC2Series electric car",
      "ThingHierarchy": {
        "HierarchyID": "582E4688A7647F2CE200229BE0434C65",
        "Name": "Car_XYZ_Hierarchy",
        "Description": "Desccribes hierarchy of units in the carXYZ",
        "ExternalID": "CarXYZ-Hierarchy",
        "RootThingID": "965B8AB45F624848BDEBD4837D093945"
      }
    }
  ]
}

```

## View immediate child nodes of a thing in a specific thing hierarchy

**Request:** /ThingHierarchy/v1/ThingHierarchyElements?\$filter=ParentThingID eq '965B8AB45F624848BDEBD4837D093945' and HierarchyID eq '582E459DA7647F2CE200229BE0434C65'&\$format=json

**Response:**

```
{
  "d": {
    "results": [
      {
        "_metadata": {
          "id": "https://appiot-thing-hierarchy-ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/ThingHierarchyElements(HierarchyID='582E459DA7647F2CE200229BE0434C65',ThingID='9EFAEFDB3F794ED0BE6ADD668D90169E')",
          "uri": "https://appiot-thing-hierarchy-ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/ThingHierarchyElements(HierarchyID='582E459DA7647F2CE200229BE0434C65',ThingID='9EFAEFDB3F794ED0BE6ADD668D90169E')",
          "type": "com.sap.appiot.ThingHierarchyElement"
        },
        "HierarchyID": "582E459DA7647F2CE200229BE0434C65",
        "ThingID": "9EFAEFDB3F794ED0BE6ADD668D90169E",
        "ThingName": "Car-XYZ-Engine",
        "ThingDescription": "Engine of Car-XYZ",
        "ParentThingID": "965B8AB45F624848BDEBD4837D093945",
        "ParentThingName": "Car-XYZ",
        "ParentThingDescription": "A car of XYZ series",
        "Description": "Describes hierarchy of units in the carXYZ",
        "Operation": null,
        "ThingType": "core.automobiles:ABC2Series",
        "ThingTypeDescription": "ABC2Series electric car",
        "ThingHierarchy": {
          "_deferred": {
            "uri": "https://appiot-thing-hierarchy-ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/ThingHierarchyElements(HierarchyID='582E459DA7647F2CE200229BE0434C65',ThingID='9EFAEFDB3F794ED0BE6ADD668D90169E')/ThingHierarchy"
          }
        }
      },
      {
        "_metadata": {
          "id": "https://appiot-thing-hierarchy-ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/ThingHierarchyElements(HierarchyID='582E459DA7647F2CE200229BE0434C65',ThingID='88E2BA6E7512439E9A771EE626AB25A2')",
          "uri": "https://appiot-thing-hierarchy-ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/ThingHierarchyElements(HierarchyID='582E459DA7647F2CE200229BE0434C65',ThingID='88E2BA6E7512439E9A771EE626AB25A2')",
          "type": "com.sap.appiot.ThingHierarchyElement"
        },
        "HierarchyID": "582E459DA7647F2CE200229BE0434C65",
        "ThingID": "88E2BA6E7512439E9A771EE626AB25A2",
        "ThingName": "Car-XYZ-ControlUnit",
        "ThingDescription": "ControlUnit of Car-XYZ",
        "ParentThingID": "965B8AB45F624848BDEBD4837D093945",
        "ParentThingName": "Car-XYZ",
        "ParentThingDescription": "A car of XYZ series",
        "Description": "Describes hierarchy of units in the carXYZ",
        "Operation": null,
        "ThingType": "core.automobiles:ABC2Series",
        "ThingTypeDescription": "ABC2Series electric car",
        "ThingHierarchy": {

```

```

      "_deferred": {
        "uri": "https://apiot-thing-hierarchy-
ci.cfapps.staging.sic.ondemand.com:443/ThingHierarchy/v1/
ThingHierarchyElements(HierarchyID='582E459DA7647F2CE200229BE0434C65',ThingID='88
E2BA6E7512439E9A771EE626AB25A2')/ThingHierarchy"
      }
    }
  ]
}

```

## Related Information

[Thing Hierarchy Element \[page 973\]](#)

## 18.1.3 Thing Hierarchy Description

Represents a thing hierarchy description

**Resource Path:** `http://<server address>[:<port number>]/ThingHierarchy/v1/ThingHierarchies?$expand=ThingHierarchyDescriptions`

## Operations

### CRUD Operations

HTTP Method	Operation	URI
<a href="#">GET</a>	<a href="#">Read Thing Hierarchy [page 956]</a>	/v1/ThingHierarchies? \$expand=ThingHierarchyDescriptions

## Thing Hierarchy Description Properties

Parameter	Data Type	Maximum Length	Description
LanguageCode	String	2	ISO code of the language of the Thing Hierarchy Description

Parameter	Data Type	Maximum Length	Description
Description	String	255	Description of the thing hierarchy.

**i Note**

The text provided in the **Description** field of the **Thing Hierarchy** entity is ignored and it is replaced with the text provided in this field.

## Related Information

[Thing Hierarchy \[page 947\]](#)

[Thing Hierarchy Element \[page 973\]](#)

## 18.2 Read Version Information of Thing Hierarchy Service

This service is used to retrieve the version information of the thing hierarchy service.

### Request

**URI:** `/ThingHierarchy/Monitor/Version`

**Operation Type:** CRUD | Action | Function

**HTTP Method:** `GET`

**Permissions:** `<ths>.r`

### Request Example

`/ThingHierarchy/Monitor/Version`



## Response

### Response Status and Error Codes

Code	Description
200	Version information retrieved successfully.

## Payload

Format: [JSON](#)

Only media type JSON is supported. The JSON for this method has the following structure:

```
{
  "d": {
    "results": [{
      "_metadata": {
        "id": "http://localhost:8183/ThingHierarchy/Monitor/Version('1.2.0-SNAPSHOT')",
        "uri": "http://localhost:8183/ThingHierarchy/Monitor/Version('1.2.0-SNAPSHOT')",
        "type": "com.sap.apiot.Version"
      },
      "Version": "1.2.0-SNAPSHOT"
    }]
  }
}
```

## 18.3 HTTP Status Codes for Thing Hierarchy

List of status codes for thing hierarchy

In the following table, you find the list of HTTP status codes that are used by the thing hierarchy services:

POST	GET (Single value)	GET (Collection)	DELETE	Description
201	200	200	204	Success
400	400	400	400	Bad request (for example: wrong filter, validation failed)
403	403	403	403	Forbidden (scope(s) not assigned for the user)
n/a	404	n/a	404	Not found
500	500	500	500	Internal server error

# 19 Event Services

Events are used to monitor and control any changes in the system that can possibly compromise safety measures and risks. You can use the event configuration service to configure event types and create events for those event types.

## Types of Events

Events can be of different types:

- Standard events: All events are mutable.
- User-defined events: Events are mutable or immutable based on the state defined for the event type.

The following table describes the state of the user-defined event type and what you can do with the corresponding events:

User-defined Event Type	EventTypeState	Events
core.automobiles:ExceededTemperature	Immutable	You cannot delete events created using this event type.
core.automobiles:EXceededPressure	Mutable	You can delete events created using this event type.

## Mutable Events

The following table illustrates the life cycle of standard events:

Method	Action	CorrelationId (user input)	Time	EventId	CorrelationId (system generated)
POST	Create		10:00:00	E1	ABCD
POST	Update	ABCD	10:02:00	E1	ABCD
POST	Create	XYZ	10:04:00	E2	XYZ
POST	Update	XYZ	10:06:00	E2	XYZ
POST	Update	XYZ	10:08:00	E2	XYZ

### Note

- Event **E1** is the latest event and the only accessible event with the correlation ID **ABCD**.
- Event **E2** is the latest event and the only accessible event with the correlation ID **XYZ**.

## Immutable Events

The following table illustrates the life cycle of predefined events:

Method	Action	CorrelationId (user input)	Time	EventId	CorrelationId (system generated)
POST	Create		10:00:00	E1	ABCD
POST	Update	ABCD	10:02:00	E2	ABCD
POST	Update	ABCD	10:04:00	E3	ABCD
POST	Create		10:06:00	E4	XYZ
POST	Update	XYZ	10:08:00	E5	XYZ
POST	Create	PQRS	10:10:00	E6	PQRS

### Note

Group of events for a specific correlation ID is used to understand the sequence of changes in the event for a specific event type.

- Events **E1**, **E2**, and **E3** are accessible with the correlation ID **ABCD**.
- Events **E4** and **E5** are accessible with the correlation ID **XYZ**.
- Event **E6** is accessible with the correlation ID **PQRS**.

## Related Information

[Standard or Predefined Events \[page 983\]](#)

[User-defined Events based on OData Event Type \[page 998\]](#)

## 19.1 Standard or Predefined Events

You can use this event service to create only standard events of predefined type (`com.sap.appiot.eventtypes:StandardEventType`) and retrieve a list of events. All standard events are mutable.

In the examples related to the industrial robots or the cars explained in the previous sections, there is information such as pressure, temperature that are constantly sent from things. This information is crucial especially when there are values that overshoot a defined threshold.

Events can be constructed such that they have a reference to a Property defined in the package. This is however, not mandatory.

### Request

#### Base URI

- **Formal description:** `http://<server address>[:<port number>][<path>]/Events`
- An example for a base URI in a cloud foundry environment: `https://apiot-mds.cfapps.eu10.hana.ondemand.com/Events`

## Methods

HTTP Method	Action	URI	Scopes
POST	<a href="#">Create an Event [page 985]</a>	<code>/Events</code>	<code>&lt;thing&gt;.Event.c</code>
GET	<a href="#">Read an Event [page 987]</a>	<code>/Events ('&lt;EventID&gt;')</code>	<code>&lt;thing&gt;.Event.r</code>
GET	<a href="#">Read all Events [page 989]</a>	<code>/Events</code>	<code>&lt;thing&gt;.Event.r</code>
POST	<a href="#">Update an Event [page 992]</a>	<code>/Events ('&lt;EventID&gt;')</code>	<code>&lt;thing&gt;.Event.u</code>
POST	<a href="#">Update Status for Multiple Events [page 993]</a>	<code>/Events.SetStatus (newStatus=&lt;status&gt;)</code>	<code>&lt;thing&gt;.Event.u</code>
DELETE	<a href="#">Delete an Event [page 994]</a>	<code>/Events ('&lt;EventID&gt;')</code>	<code>&lt;thing&gt;.Event.d</code>
DELETE	<a href="#">Delete all Events [page 995]</a>	<code>/Events&amp;?filter=&lt;filter&gt; AND _businessTimeStamp &lt;operator&gt; &lt;timestamp&gt;</code>	<code>&lt;thing&gt;.Event.d</code>
GET	<a href="#">Read Version Information of Event Service [page 996]</a>	<code>/Version</code>	<code>&lt;thing&gt;.Event.r</code>

### Note

`<thing>` refers to the thing application name.

## Related Information

[Event Services \[page 982\]](#)

## 19.1.1 Create an Event

This service is used to create an event of predefined event type  
`com.sap.appiot.eventtypes:StandardEventType`.

### Request

**URI:** `/Events`

**HTTP Method:** *POST*

**Permissions:**

`<thing>.Event.c`

### Request Example

`/Events`

### Payload

```
{
  "_businessTimeStamp": "2015-10-29T09:00:00Z",
  "_status": "Open",
  "_type": "Alert",
  "_severity": 1,
  "_source": "Test Source111",
  "_code": "1",
  "_thingId": "965B8AB45F624848BDEBD4837D093945",
  "_thingProperty": "core.automobiles:ABCXSeries/wheel_fr",
  "_description": "Front right wheel worn out",
  "_externalId": "AA2X-EXT"
}
```

### Request Properties

#### Event Object Properties

Property	Type	Mandatory	Maximum Length	Description
<code>_id</code>	string	No	32	Internal identifier for an Event.
<code>_businessTimeStamp</code>	string	No	24	Business timestamp corresponding to the event that is created.

Property	Type	Mandatory	Maximum Length	Description
_status	string	Yes	50	Status of the event.  <div> <b>i Note</b>  The following are the only supported statuses for standard events: <ul style="list-style-type: none"> <li>• Open</li> <li>• InProcess</li> <li>• Completed</li> </ul> </div>
_severity	integer	No	-	Severity of the event.  <div> <b>i Note</b>  The following are the only supported severities for standard events: <ul style="list-style-type: none"> <li>• 1</li> <li>• 2</li> <li>• 3</li> </ul> </div>
_type	string	No	71	Type of the event.
_source	string	No	100	Source of the event.
_code	string	Yes	50	Code for the event.
_thingId	string	Yes	32	Thing ID corresponding to the event created
_thingProperty	string	No	255	Thing property corresponding to the event created
_description	string	No	200	Message text used to describe the event.
_externalId	string	No	255	Event identifier (which is not unique) from an external system. You can create multiple events with the same external ID.  <div> <b>i Note</b>  Only the following characters are supported: <ul style="list-style-type: none"> <li>• Uppercase and lowercase alphabets <i>a</i> through <i>z</i> and <i>A</i> through <i>Z</i></li> <li>• Numeric digits <i>0</i> through <i>9</i></li> <li>• Punctuation marks hyphen (-), underscore (_), colon (:), and full-stop (.)</li> </ul> </div>

## Response

Format: *JSON*

## Response Status and Error Codes

Code	Description
201	Event created successfully

### Note

The URL of the event created is returned in the location header of the response.

## Related Information

[Standard or Predefined Events \[page 983\]](#)

## 19.1.2 Read an Event

This service is used to retrieve the details of a specified event.

## Request

URI: /Events ('<Event ID>')

HTTP Method: [GET](#)

Permissions:

<thing>.Event.r

## Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header CONTENT-ENCODING with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Parameter

Parameter	Required	Data Type	Description
id	Yes	string	Internal identifier for an event.

## Request Example

```
/Events('<EventID>')
```

## Response

Format: [JSON](#)

## Response Status and Error Codes

Code	Description
200	Event details retrieved successfully

## Payload

Format: [JSON](#)

Only media type [JSON](#) is supported. The [JSON](#) for this method has the following structure:

```
{
  "_id": "<eventID>",
  "_businessTimeStamp": "2015-10-29T09:00:00Z",
  "_status": "Open",
  "_type": "Alert",
  "_severity": 1,
  "_source": "Test Source111",
  "_code": "1",
  "_thingId": "965B8AB45F624848BDEBD4837D093945",
  "_thingProperty": "core.automobiles:ABCXSeries/wheel_fr",
  "_description": "Front right wheel worn out",
  "_externalId": "AA2X-EXT"
}
```

## Related Information

[Standard or Predefined Events \[page 983\]](#)



## 19.1.3 Read all Events

This service is used to read all the existing events.

### i Note

Retrieving number of objects more than the recommended limits with only one collective GET request can lead to high system load and a significant increase of response times.

To avoid performance decline, the system in future versions will automatically limit collective GET requests to a maximum of 500 objects that can be retrieved with one single call.

Use of `$count` as well degrades performance. Hence, a use of `$count` must also be judiciously chosen.

Therefore, for use cases where it is crucial for you to ensure that objects more than 500 have to be retrieved, you need to use paging options such as `$top` and `$skip`.

## Request

URI: `/Events`

HTTP Method: `GET`

Permissions:

`<thing>.Event.r`

## Request Headers

Parameter	Required	Description
<code>Accept-Encoding</code>	No	Retrieve compressed response  Only if you specify the value <code>gzip</code> or <code>gzip,deflate</code> for this parameter, the system returns the compressed response with an additional response header <code>CONTENT-ENCODING</code> with value <code>gzip</code> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Parameters

None

## Query String Parameters

Parameter	Required	Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set.

Parameter	Required	Type	Description
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$count	No	Boolean	<p>Total number of event objects available based on the current user's authorizations; permissible values are <i>true</i> and <i>false</i>.</p> <p>The following results apply, if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"> <li>• If you use this with the \$top and \$skip parameters, the result contains the total count of event objects available.</li> <li>• If you use this with \$filter parameter, the result contains the total count of event objects based on the defined filter condition.</li> </ul>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; valid fields are <i>_status</i>, <i>_severity</i>, and <i>_businessTimeStamp</i>.</p> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>
\$filter	No	String	Filter condition to be applied; valid fields are <i>_thingId</i> , <i>_thingProperty</i> , <i>_status</i> and <i>_businessTimeStamp</i> . The filter operators supported are <i>eq</i> , <i>gt</i> , <i>le</i> , <i>ge</i> , and <i>ne</i> .
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are <i>_thingId</i> , <i>_thingProperty</i> , <i>_status</i> , and <i>_businessTimeStamp</i> .

## Request Example

```
/Events?$filter=_thingProperty eq 'core.automobiles:ABCXSeries/wheel_fr'
```

Retrieves all events where the thing property is `core.automobiles:ABCXSeries/wheel_fr`.

```
/Events?$filter=_thingProperty eq 'core.automobiles:ABCXSeries/wheel_fr'&
$select=_thingProperty, _status
```

Retrieves all events where the thing property is `core.automobiles:ABCXSeries/wheel_fr`. However, with \$select parameter, only the fields *\_thingProperty* and *\_status* are displayed for all the retrieved events.

## Response

Format: *JSON*

## Response Properties

Property	Type	Description
value	Array of event objects	Array of event objects (Based on whether there are filter conditions applied.).

## Response Status and Error Codes

Code	Description
200	Events retrieved successfully

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  value: [
    {
      "_id": "<eventID>",
      "_businessTimeStamp": "2015-10-29T09:00:00Z",
      "_status": "Open",
      "_type": "Alert",
      "_severity": 1,
      "_source": "Test Source111",
      "_code": "1",
      "_thingId": "965B8AB45F624848BDEBD4837D093945",
      "_thingProperty": "core.automobiles:ABCXSeries/wheel_fr",
      "_description": "Front right wheel worn out",
      "_externalId": "AA2X-EXT"
    },
    {
      "_id": "<eventID2>",
      "_businessTimeStamp": "2015-10-29T09:00:00Z",
      "_status": "In progress",
      "_type": "Alert",
      "_severity": 3,
      "_source": "Test Source111",
      "_code": "1",
      "_thingId": "<thingID>",
      "_thingProperty": "core.automobiles:ABCXSeries/wheel_rr",
      "_description": "Rear right wheel worn out",
      "_externalId": "AA2X-EXT"
    }
  ]
}
```

## Related Information

[Standard or Predefined Events \[page 983\]](#)

## 19.1.4 Update an Event

This service is used to update certain attributes associated with an event.

### Request

URI: /Events('<Event ID>')

HTTP Method: *POST*

Permissions:

<thing>.Event.u

### Request Parameter

Parameter	Required	Data Type	Description
id	Yes	string	Internal identifier for an event.

### Request Example

```
{
  "_businessTimeStamp": "2015-10-29T09:00:00Z",
  "_status": "Open",
  "_type": "Alert",
  "_severity": 2,
  "_source": "Test Source111",
  "_code": "1",
  "_thingId": "<thingID>",
  "_thingProperty": "core.automobiles:ABCXSeries/wheel_fr",
  "_description": "Front right wheel worn out",
  "_externalId": "AA2X-EXT"
}
```

### Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
200	Attributes of the specified event updated successfully

### Related Information

[Standard or Predefined Events \[page 983\]](#)

## 19.1.5 Update Status for Multiple Events

This service is used to update the status for multiple events.

### Request

**URI:** `/Events.SetStatus(newStatus=<status>)`

**HTTP Method:** *POST*

**Permissions:**

`<thing>.Event.u`

### Request Example

```
{
  "events": [{
    "_id": "<EventID1>"
  }, {
    "_id": "<EventID2>"
  }]
}
```

### Response

**Format:** *JSON*

### Response Status and Error Codes

Code	Description
200	Statuses of all events updated successfully.

### Response Example

```
200-OK
```

### Related Information

[Standard or Predefined Events \[page 983\]](#)

## 19.1.6 Delete an Event

This service is used to delete an event with specific event ID. This service supports deletion of standard events of event type `com.sap.appiot.eventtypes:StandardEventType`.

### Request

URI: `/Events('<Event ID>')`

HTTP Method: `DELETE`

Permissions:

`<thing>.Event.d`

### Request Parameter

Parameter	Required	Data Type	Description
<code>id</code>	Yes	string	Internal identifier for an event.

### Request Example

```
/Events('<EventID>')
```

### Response

Format: `JSON`

### Response Status and Error Codes

Code	Description
200	Event deleted successfully

### Related Information

[Standard or Predefined Events \[page 983\]](#)

## 19.1.7 Delete all Events

This service is used to delete the required events.

### Request

**URI:** /Events?\$filter=<field> <operator> <value> and \_businessTimeStamp <operator> <timestamp>

**HTTP Method:** *DELETE*

**Permissions:**

<thing>.Event.d

### Request Parameters

None

### Request Example

/Events?\$filter=\_thingProperty eq 'core.automobiles:ABCXSeries/wheel\_fr' and \_businessTimeStamp lt 2016-01-01T09:01:00Z

### Query String Parameters

Parameter	Type	Mandatory	Description
\$filter	String	Yes	Filter condition to be applied; valid fields are <i>_businessTimeStamp</i> , <i>_thingId</i> , <i>_status</i> and <i>_thingProperty</i> .

**i Note**  
*\_businessTimeStamp* is a mandatory field.

### Response

**Format:** *JSON*

### Response Status and Error Codes

Code	Description
200	Events deleted successfully

### Note

The events that belong to a thing for which you have the capabilities assigned are deleted. Other events that belong to a thing for which you do not have the capabilities assigned are not deleted. Whether the events are deleted or not deleted, the status code displayed is 200.

## Related Information

[Standard or Predefined Events \[page 983\]](#)

## 19.1.8 Read Version Information of Event Service

This service is used to retrieve the version information of the event service.

### Request

URI: `/Version`

HTTP Method: `GET`

Permissions:

`<thing>.Event.r`

### Request Example

```
/Version
```

### Response

Format: `JSON`

### Response Status and Error Codes

Code	Description
200	Version information retrieved successfully.

### Response Example

Format: `JSON`



Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
{
  "version" : "1.10"
}
```

## Related Information

[Standard or Predefined Events \[page 983\]](#)

### 19.1.9 HTTP Status Codes for Event Services

List of status codes for Event services

In the following table, you find the list of HTTP status codes that are used by the event services:

Post (Create)	Post (Update)	Put	Delete	Get (single value)	Get (collection)	Description
201	200	200	204	200	200	Success
n/a	404	404	n/a	404	200	No capabilities assigned
201	200	200	200	404	200	Read capability not assigned to read a specific thing. However, the user is assigned with the capabilities to perform operations such as create, update, and delete.
404	404	404	404	200	200	Read capability assigned to read a specific thing. However, the user is not assigned with capabilities to perform operations such as create, update, and delete.
n/a	404	404	404	404	n/a	Not found. The requested thing is not found.

Post (Create)	Post (Update)	Put	Delete	Get (single value)	Get (collection)	Description
403	403	403	403	403	403	Forbidden (scope(s) not assigned for the user)
400	400	400	400	400	400	Bad request (for example: header field missing, wrong filter, validation failed)
500	500	500	500	500	500	Internal server error

## Related Information

[Standard or Predefined Events \[page 983\]](#)

## 19.2 User-defined Events based on OData Event Type

In this section, you will learn about events based on the event type configured using the OData services.

**Resource Path:** `/ES/EventType/<event type name>/v1/Events`

**Example for a base URI in a cloud foundry environment:** `https://events-sap.cfapps.eu10.hana.ondemand.com/ES/EventType/<event type name>/v1/Events`

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scope
<i>POST</i>	<a href="#">Create Events [page 1001]</a>	<code>/ES/EventType/&lt;event type name&gt;/v1/Events</code>	<code>&lt;tde&gt;.c</code>
<i>GET</i>	<a href="#">Read Events and Event Data [page 1008]</a>	<code>/ES/EventType/&lt;event type name&gt;/v1/Events</code>	<code>&lt;tde&gt;.r</code>
<i>DELETE</i>	<a href="#">Delete an Event [page 1013]</a>	<code>/ES/EventType/&lt;event type name&gt;/v1/Events('&lt;event ID&gt;')</code>	<code>&lt;tde&gt;.d</code>

HTTP Method	Operation	URI	Scope
GET	<a href="#">Read Event Correlations [page 1025]</a>	/ES/v1/EventCorrelations	<td>.r

## Related Information

[Event Configuration \[page 512\]](#)

## 19.2.1 Read Metadata for Event of an Event Type

Retrieves metadata for events

With this service, you can retrieve metadata for an event that belongs to a specific event type.

### Request

**URI:** /ES/EventType/<event type name>/v1/\$metadata

**Operation Type:** CRUD

**HTTP Method:** GET

**Permissions:** <td>.r

### Request Parameters

None

### Request Example

/ES/EventType/core.automobiles:ExceededTemperature/v1/\$metadata

### Response

#### Response Status and Error Codes

Code	Description
200	Metadata retrieved successfully

## Payload

Format: *XML*

### Example 1

/ES/EventType/core.automobiles:ExceededTemperature/v1/\$metadata

```
<EntityType Name="DYN_ENT_core_automobiles__ExceededTemperatures">
  <Key>
    <PropertyRef Name="EventId"></PropertyRef>
  </Key>
  <Property Name="EventId" Type="Edm.String"></Property>
  <Property Name="eventData.Temperature" Type="Edm.String"></
Property>
</EntityType>
<EntityType Name="Events">
  <Key>
    <PropertyRef Name="EventId"></PropertyRef>
  </Key>
  <Property Name="EventId" Type="Edm.String" Nullable="false"></
Property>
  <Property Name="CorrelationId" Type="Edm.String"></Property>
  <Property Name="BusinessTimestamp" Type="Edm.String"></Property>
  <Property Name="Type" Type="Edm.String"></Property>
  <Property Name="EventType" Type="Edm.String"></Property>
  <Property Name="EventStatus" Type="Edm.String"></Property>
  <Property Name="EventSeverity" Type="Edm.Int64"></Property>
  <Property Name="EventCode" Type="Edm.String"></Property>
  <Property Name="EventSource" Type="Edm.String"></Property>
  <Property Name="ThingId" Type="Edm.String"></Property>
  <Property Name="ThingProperty" Type="Edm.String"></Property>
  <Property Name="ExternalId" Type="Edm.String"></Property>
  <Property Name="EventInfo" Type="Edm.String"></Property>
  <NavigationProperty
Name="DYN_ENT_core_automobiles__ExceededTemperatures"
Relationship="com.sap.apptot.es.DYN_ENT_core_automobiles__ExceededTemperatures"
FromRole="EventEntities"
ToRole="DYN_ENT_core_automobiles__ExceededTemperatureEntities"></
NavigationProperty>
</EntityType>
<Association Name="DYN_ENT_core_automobiles__ExceededTemperatures">
  <End Type="com.sap.apptot.es.Event" Multiplicity="1"
Role="EventEntities"></End>
  <End
Type="com.sap.apptot.es.DYN_ENT_core_automobiles__ExceededTemperature"
Multiplicity="1" Role="DYN_ENT_core_automobiles__ExceededTemperatureEntities"></
End>
</Association>
<EntityContainer Name="UnifiedEventsEventType"
m:IsDefaultEntityContainer="true">
  <EntitySet Name="DYN_ENT_core_automobiles__ExceededTemperatures"
EntityType="com.sap.apptot.es.DYN_ENT_core_automobiles__ExceededTemperature"></
EntitySet>
  <EntitySet Name="Events" EntityType="com.sap.apptot.es.Event"></
EntitySet>
  <AssociationSet
Name="DYN_ENT_core_automobiles__ExceededTemperatures"
Association="com.sap.apptot.es.DYN_ENT_core_automobiles__ExceededTemperatures">
    <End EntitySet="Events" Role="EventEntities"></End>
  <End
EntitySet="DYN_ENT_core_automobiles__ExceededTemperatures"
Role="DYN_ENT__core_automobiles__ExceededTemperatureEntities"></End>
  </AssociationSet>
</EntityContainer>
```

## Related Information

[User-defined Events based on OData Event Type \[page 998\]](#)

### 19.2.2 Create Events

With this method, you can create standard events or user-defined events along with the event data.

It is mandatory, that you must specify a valid event type (user-defined or standard event type) name in the request URL. The service creates an event only for the event type name specified in the request URL. If the event type names specified in the request URL and the request payload are different, the service displays an error message and no event is created. For example, the following are the request URLs to create different kinds of events:

- `/ES/EventType/core.automobiles:ExceededTemperature/v1/Events`
  - Along with a relevant request payload, you can create events of user-defined event type `core.automobiles:ExceededTemperature`
- `/ES/EventType/com.sap.apptot.eventtypes:StandardEventType/v1/Events` or `/ES/EventType/<randomtext>/Events`
  - Along with a relevant request payload, you can create standard events of predefined event type `com.sap.apptot.eventtypes:StandardEventType`

All events are associated with a correlation ID. The correlation ID is used to group all related events and define the life cycle of an event. The correlation ID is an optional field. Multiple events can share the same correlation ID. However, events belonging to different event types cannot use the same correlation ID.

#### i Note

- The `POST` method supports both creation of a new event and update of an existing event.
- When you specify the correlation ID in the request payload of a `POST` method, the system validates the correlation ID and performs the following:
  - If an immutable event already exists with the correlation ID and you are creating an immutable event, the system creates a new event with a unique event ID and assigns the specified correlation ID.
  - If a mutable event already exists with the correlation ID and you are creating a mutable event, the system updates the existing event with new values for the fields provided in the request payload. After this update, you cannot access the original event that is updated.
  - If an event does not exist with the correlation ID and you are creating a mutable or immutable event, the system creates a new event with a unique event ID and assigns the specified correlation ID.
- When you do not specify the correlation ID in the request payload of a `POST` method, the system creates new event with unique event ID and generates a unique 32 bit GUID as correlation ID.

When you create an event, the fields [EventStatus](#), [EventSeverity](#), and [Eventcode](#) are mandatory. However, if you have not defined enumeration values for these fields while creating the event type configuration, you can assign any free text for these fields. If you have defined enumeration values for these fields while creating the event type configuration, you can assign only the defined values for these fields.

## Request

**URI:** /ES/EventType/<event type name>/v1/Events

**Operation Type:** CRUD

**HTTP Method:** *POST*

**Permissions:** <td>.c

### Request Parameters

Parameter	Required	Data Type	Description
EventType	Yes	String	Unique name of the event type

### Event Object Properties

Property	Type	Mandatory	Maximum Length	Description
CorrelationId	String	No	32	Unique identifier to group all related events
BusinessTimestamp	String	Yes	24	Business timestamp corresponding to the event that is created
Type	String	No	81	Additional information about the event
EventType	String	No		User-defined event type name
EventStatus	String	Yes	50	Status of the event

#### i Note

The following are the only supported statuses for standard events:

- Open
- InProcess
- Completed

Property	Type	Mandatory	Maximum Length	Description
EventSeverity	Integer	Yes		Severity of the event
				<b>i Note</b> The following are the only supported severities for standard events: <ul style="list-style-type: none"> <li>• 1</li> <li>• 2</li> <li>• 3</li> </ul>
EventCode	String	No	50	Code of the event
EventSource	String	No	100	Source of the event
ThingId	String	Yes		Unique identification of a thing for which an event is created
ThingProperty	String	No	255	Property of a thing for which the event is created
				<b>→ Recommendation</b> We recommend that you specify <thing type name>/<property set ID>. For example, you can specify <code>core.automobiles:ABC2Series/wheel_rr.</code>

Property	Type	Mandatory	Maximum Length	Description
ExternalId	String	No	255	Event identifier (which is not unique) from an external system. You can create multiple events with the same external ID.
				<b>i Note</b> Only the following characters are supported: <ul style="list-style-type: none"> <li>• Uppercase and lowercase alphabets a through z and A through Z</li> <li>• Numeric digits 0 through 9</li> <li>• Punctuation marks hyphen (-), underscore (_), colon (:), and full-stop (.)</li> </ul>
EventInfo	String	No	200	Free text to provide additional information about the event.

## Payload

**Format:** *JSON*

Media types *JSON* and *XML* are supported. The *JSON* for this method has the structure as depicted in the following examples:

### Example 1

The following request URL, along with the payload, creates an event of user-defined event type `core.automobiles:ExceededTemperature` along with the event data.

`/ES/EventType/core.automobiles:ExceededTemperature/v1/Events`

```
{
  "BusinessTimestamp": "2015-08-01T09:00:00Z",
  "Type": "Alert",
  "EventType": "core.automobiles:ExceededTemperature",
```



```

    "EventInfo": "Alert on temperature",
    "EventStatus": "Open",
    "EventSeverity": 1,
    "EventCode": null,
    "EventSource": null,
    "ThingId": "4D622E3DA6A14C4D87CDEC935B000001",
    "ThingProperty": "Engine Temperature",
    "ExternalId": "SampleEvent",
    "DYN_ENT_com_sap_automobiles__TemperatureEventPST": {
      "Temperatrure": "104.0"
    }
  }
}

```

## Example 2

The following request URL, along with the payload, creates a standard event of pre-defined event type `com.sap.appiot.eventtypes:StandardEventType` along with the event data.

`/ES/EventType/com.sap.appiot.eventtypes:StandardEventType/v1/Events`

```

{
  "BusinessTimestamp": "2015-08-01T09:00:00Z",
  "Type": "Alert",
  "EventInfo": "Alert on engine temperature",
  "EventStatus": "Open",
  "EventSeverity": 1,
  "EventCode": null,
  "EventSource": null,
  "ThingId": "4D622E3DA6A14C4D87CDEC935B000001",
  "ThingProperty": "Engine Temperature",
  "ExternalId": "SampleEvent"
}

```

## Response

### Response Status and Error Codes

Code	Description
201	Event created successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as illustrated in the following examples.

The following is the response payload after creating an event of user-defined event type `core.automobiles:ExceededTemperature`.

```

{
  "d": {
    "__metadata": {

```

```

      "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/
EventType/core.automobiles:ExceededTemperature/v1/
Events('F392018DE4D54CCFB5FEC72672D4310B')",
      "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/
EventType/core.automobiles:ExceededTemperature/v1/
Events('F392018DE4D54CCFB5FEC72672D4310B')",
      "type": "com.sap.apptot.es.Events"
    },
    "EventId": "F392018DE4D54CCFB5FEC72672D4310B",
    "CorrelationId": "BD622E3DA6A14C4D87CDEC935B000001",
    "BusinessTimestamp": "/Date(1438419600000)/",
    "Type": "Alert",
    "EventType": "core.automobiles:ExceededTemperature",
    "EventInfo": "Alert on temperature",
    "EventStatus": "Open",
    "EventSeverity": "1",
    "EventCode": null,
    "EventSource": null,
    "ThingId": "4D622E3DA6A14C4D87CDEC935B000001",
    "ThingProperty": "Engine temperature",
    "ExternalId": "SampleEvent",
    "DYN_ENT_com_sap_automobiles__TemperatureEventPSTs": {
      "deferred": {
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/EventType/core.automobiles:ExceededTemperature/v1/
Events('F392018DE4D54CCFB5FEC72672D4310B')/
DYN_ENT_com_sap_automobiles__TemperatureEventPSTs"
      }
    }
  }
}

```

## Payload Validation

The payload is validated before updating the data in the database. An appropriate error message is displayed to correct the payload for the event data.

**Header:** sap-iot-strong-validations

**Value:** TRUE

The following table describes the payload validations for the different value data types:

Data Types	Action
<i>LargeString</i> , <i>JSON</i> , and <i>ByteArray</i> along with invalid property names	Always validated by default (regardless of the current setting of sap-iot-strong-validations).
<i>String</i> , <i>Numeric</i> , <i>Boolean</i> , <i>ThingID</i> , <i>BPID</i> , <i>Timestamp</i> , <i>DateTime</i> , and <i>Date</i> , along with <i>JSON</i> , <i>ByteArray</i> along with invalid property names	Validated only if the header parameter sap-iot-strong-validations is set to TRUE in the request.

### Note

SAP Leonardo IoT supports three different date time format - *Date*, *TimeStamp*, and *DateTime*. If the header parameter sap-iot-strong-validations is set to FALSE, the value for the property of these data types is not validated for the supported date time format. For example, for a property of data type

Date, the service allows you to specify the value in *TimeStamp* format. However, while retrieving the data, the service displays the data in the correct format of the data type, that is *Date*.

The following is an example of invalid payload:

```
{
  "BusinessTimestamp": "2015-08-01T09:00:00Z",
  "Type": "Alert",
  "EventType": "core.automobiles:ExceededSpeed",
  "EventInfo": "Alert on speed",
  "EventStatus": "Open",
  "EventSeverity": 1,
  "EventCode": null,
  "EventSource": null,
  "ThingId": "4D622E3DA6A14C4D87CDEC935B000001",
  "ThingProperty": "Engine Speed",
  "ExternalId": "SampleEvent",
  "DYN_ENT_com_sap_automobiles__SpeedEventPST": {
    "Speed": "Fast"
  }
}
```

## Response Status and Error Codes

Code	Description
400	Bad request

## Response Example

```
{
  "error": {
    "code": null,
    "message": {
      "lang": "en",
      "value": "{ \"messages\": { \"appiot.invalidNumeric\": \"Invalid value 'Fast' for property Speed of type Numeric.\" }, \"correlationId\": \"0af05cba-11c7-473c-4a85-032e4e9608b5\" }"
    }
  }
}
```

The following is an example of valid payload:

## Request Example

```
{
  "BusinessTimestamp": "2015-08-01T09:00:00Z",
  "Type": "Alert",
  "EventType": "core.automobiles:ExceededSpeed",
  "EventInfo": "Alert on speed",
  "EventStatus": "Open",
  "EventSeverity": 1,
  "EventCode": null,
  "EventSource": null,
  "ThingId": "4D622E3DA6A14C4D87CDEC935B000001",
  "ThingProperty": "Engine Speed",
  "ExternalId": "SampleEvent",
  "DYN_ENT_com_sap_automobiles__SpeedEventPST": {
    "Speed": "340"
  }
}
```

## Related Information

[User-defined Events based on OData Event Type \[page 998\]](#)

### 19.2.3 Read Events and Event Data

With this method you can retrieve all events that belong to a specific event type. The response payload includes event data as well.

In addition, you can retrieve a subset of all events, according to the filter criteria provided. For a collective request for a set of events, you have these options:

- You can restrict the set of events to be retrieved by filter criteria based on all fields available in the response.
- You can define that the set of matching events shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ES/EventType/<event type name>/v1/Events`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<tde>.r`

## Request Parameters

Parameter	Required	Data Type	Description
EventType	Yes	String	Name of the event type <ul style="list-style-type: none"><li>• Specify the user-defined event type name to retrieve events of user-defined event type</li><li>• Specify <code>com.sap.appiot.eventtypes:StandardEventType</code> to retrieve standard events</li></ul>

## Query Request Parameters

Parameter	Type	Required	Description
\$top	Integer	No	Number of records to include in the result set.
\$skip	Integer	No	Number of records to exclude from the result set.
\$orderby	String	No	Field name to be used for sorting the result set in ascending order; valid fields are <a href="#">CorrelationId</a> , <a href="#">BusinessTimestamp</a> , <a href="#">EventType</a> , <a href="#">EventStatus</a> , <a href="#">EventSeverity</a> , <a href="#">EventCode</a> , <a href="#">ThingId</a> , <a href="#">ThingProperty</a> , and <a href="#">ExternalId</a> .  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$select	String	No	Select condition for the data fields to be included in the result set; valid fields are <a href="#">CorrelationId</a> , <a href="#">BusinessTimestamp</a> , <a href="#">EventType</a> , <a href="#">EventStatus</a> , <a href="#">EventSeverity</a> , <a href="#">EventCode</a> , <a href="#">ThingId</a> , <a href="#">ThingProperty</a> , and <a href="#">ExternalId</a> .
\$filter	String	No	Filter condition to be applied; all fields in the result set are valid.

## Request Example

/ES/EventType/core.automobiles:ExceededTemperature/v1/Events

Retrieves all events of user-defined event type `core.automobiles:ExceededTemperature`

/ES/EventType/com.sap.appiot.eventtypes:StandardEventType/v1/Events

Retrieves all events of pre-defined event type `com.sap.appiot.eventtypes:StandardEventType`

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: *JSON*

Media types *JSON* and *XML* are supported. The *JSON* for this method has the structure as depicted in the following examples:

### Example 1

/ES/EventType/core.automobiles:ExceededTemperature/v1/Events

Retrieves static data along with dynamic data for all events of user-defined event type  
core.automobiles:ExceededTemperature

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/core.automobiles:ExceededTemperature/v1/Events('10A1FA6AA5C64812B24B7280E1E40FEE')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/core.automobiles:ExceededTemperature/v1/Events('10A1FA6AA5C64812B24B7280E1E40FEE')",
        "type": "com.sap.apptot.es.Events"
      },
      "EventId": "F392018DE4D54CCFB5FEC72672D4310B",
      "CorrelationId": "BD622E3DA6A14C4D87CDEC935B000001",
      "BusinessTimestamp": "2018-08-01T09:00:00Z",
      "Type": "Alert",
      "EventType": "core.automobiles:ExceededTemperature",
      "EventInfo": "Alert on temperature",
      "EventStatus": "Open",
      "EventSeverity": "1",
      "EventCode": "EQ12",
      "EventSource": "",
      "ThingId": "001906DB14B54326A5FA0C56BD53212D",
      "ThingProperty": "Engine temperature",
      "ExternalId": "SampleEvent",
      "DYN_ENT_core_automobiles__Engines": {
        "__deferred": {
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/core.automobiles:ExceededTemperature/v1/Events('10A1FA6AA5C64812B24B7280E1E40FEE')/DYN_ENT_core_automobiles__Engines"
        }
      }
    }],
    {
      "__metadata": {
```

```

        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/EventType/core.automobiles:ExceededTemperature/v1/
Events('E2B6A783639D4DEB96B744CCD3C8652C')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/EventType/core.automobiles:ExceededTemperature/v1/
Events('E2B6A783639D4DEB96B744CCD3C8652C')",
        "type": "com.sap.apptot.es.Events"
    },
    "EventId": "E2B6A783639D4DEB96B744CCD3C8652C",
    "CorrelationId": "BD622E3DA6A14C4D87CDEC935B000001",
    "BusinessTimestamp": "2018-08-01T09:00:00Z",
    "Type": "Alert",
    "EventType": "core.automobiles:ExceededTemperature",
    "EventInfo": "Alert on temperature",
    "EventStatus": "InProgress",
    "EventSeverity": "1",
    "EventCode": "EQ12",
    "EventSource": "",
    "ThingId": "001906DB14B54326A5FA0C56BD53212D",
    "ThingProperty": "Engine temperature",
    "ExternalId": "SampleEvent",
    "DYN_ENT_core_automobiles__Engines": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/
core.automobiles:ExceededTemperature/v1/
Events('E2B6A783639D4DEB96B744CCD3C8652C')/DYN_ENT_core_automobiles__Engines"
        }
    }
},
{
    "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/EventType/core.automobiles:ExceededTemperature/v1/
Events('F2A5A34034CE4B0DB2BCD10DD30F07C2')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/EventType/core.automobiles:ExceededTemperature/v1/
Events('F2A5A34034CE4B0DB2BCD10DD30F07C2')",
        "type": "com.sap.apptot.es.Events"
    },
    "EventId": "F2A5A34034CE4B0DB2BCD10DD30F07C2",
    "CorrelationId": "BD622E3DA6A14C4D87CDEC935B000001",
    "BusinessTimestamp": "2018-08-01T09:00:00Z",
    "Type": "Alert",
    "EventType": "core.automobiles:ExceededTemperature",
    "EventInfo": "Alert on temperature",
    "EventStatus": "Completed",
    "EventSeverity": "1",
    "EventCode": "EQ12",
    "EventSource": "",
    "ThingId": "001906DB14B54326A5FA0C56BD53212D",
    "ThingProperty": "Engine temperature",
    "ExternalId": "SampleEvent",
    "DYN_ENT_core_automobiles__Engines": {
        "__deferred": {
            "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/
core.automobiles:ExceededTemperature/v1/
Events('F2A5A34034CE4B0DB2BCD10DD30F07C2')/DYN_ENT_core_automobiles__Engines"
        }
    }
}
]
}
}

```

## Example 2

/ES/EventType/core.bbautomobiles:ExceededTemperature/v1/Events?

\$expand=DYN\_ENT\_core\_automobiles\_\_Engines

Retrieves all events of user-defined event type `core.automobiles:ExceededTemperature` along with details of dynamic properties of the event property set type assigned to the user-defined event type.

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/core.automobiles:ExceededTemperature/v1/Events('10A1FA6AA5C64812B24B7280E1E40FEE')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/core.automobiles:ExceededTemperature/v1/Events('10A1FA6AA5C64812B24B7280E1E40FEE')",
        "type": "com.sap.apptot.es.Events"
      },
      "EventId": "F392018DE4D54CCFB5FEC72672D4310B",
      "CorrelationId": "BD622E3DA6A14C4D87CDEC935B000001",
      "BusinessTimestamp": "2018-08-01T09:00:00Z",
      "Type": "Alert",
      "EventType": "core.automobiles:ExceededTemperature",
      "EventInfo": "Alert on temperature",
      "EventStatus": "Open",
      "EventSeverity": "1",
      "EventCode": "EQ12",
      "EventSource": null,
      "ThingId": "001906DB14B54326A5FA0C56BD53212D",
      "ThingProperty": "Engine temperature",
      "ExternalId": "SampleEvent",
      "DYN_ENT_core_automobiles__Engines": {
        "EventId": "10A1FA6AA5C64812B24B7280E1E40FEE",
        "eventData.Speed2": 0,
        "eventData.Temperature": "104.0"
      }
    }],
    {
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/core.automobiles:ExceededTemperature/v1/Events('E2B6A783639D4DEB96B744CCD3C8652C')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/core.automobiles:ExceededTemperature/v1/Events('E2B6A783639D4DEB96B744CCD3C8652C')",
        "type": "com.sap.apptot.es.Events"
      },
      "EventId": "E2B6A783639D4DEB96B744CCD3C8652C",
      "CorrelationId": "BD622E3DA6A14C4D87CDEC935B000001",
      "BusinessTimestamp": "2018-08-01T09:00:00Z",
      "Type": "Alert",
      "EventType": "core.automobiles:ExceededTemperature",
      "EventInfo": "Alert on temperature",
      "EventStatus": "Open",
      "EventSeverity": "1",
      "EventCode": "EQ12",
      "EventSource": null,
      "ThingId": "001906DB14B54326A5FA0C56BD53212D",
      "ThingProperty": "Engine temperature",
      "ExternalId": "SampleEvent",
      "DYN_ENT_core_automobiles__Engines": {
        "EventId": "E2B6A783639D4DEB96B744CCD3C8652C",
        "eventData.Speed2": 0,
        "eventData.Temperature": "104.0"
      }
    }
  ],
  {

```



```

        "__metadata": {
          "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/core.automobiles:ExceededTemperature/v1/Events('F2A5A34034CE4B0DB2BCD10DD30F07C2')",
          "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/EventType/core.automobiles:ExceededTemperature/v1/Events('F2A5A34034CE4B0DB2BCD10DD30F07C2')",
          "type": "com.sap.apptot.es.Events"
        },
        "EventId": "F2A5A34034CE4B0DB2BCD10DD30F07C2",
        "CorrelationId": "BD622E3DA6A14C4D87CDEC935B000001",
        "BusinessTimestamp": "2018-08-01T09:00:00Z",
        "Type": "Alert",
        "EventType": "core.automobiles:ExceededTemperature",
        "EventInfo": "Alert on temperature",
        "EventStatus": "Open",
        "EventSeverity": "1",
        "EventCode": "EQ12",
        "EventSource": null,
        "ThingId": "001906DB14B54326A5FA0C56BD53212D",
        "ThingProperty": "Engine temperature",
        "ExternalId": "SampleEvent",
        "DYN_ENT_core_automobiles_Engines": {
          "EventId": "F2A5A34034CE4B0DB2BCD10DD30F07C2",
          "eventData.Speed2": 0,
          "eventData.Temperature": "104.0"
        }
      }
    ]
  }
}

```

## Related Information

[User-defined Events based on OData Event Type \[page 998\]](#)

## 19.2.4 Delete an Event

With this method, you can delete an event with the specific event ID.

### Note

You can only delete mutable events and standard events. The service does not allow you to delete immutable events.

## Request

**URI:** /ES/EventType/<event type name>/v1/Events('<event ID>')

**Operation Type:** CRUD

**HTTP Method:** *DELETE*

**Permissions:** <td>.d

## Request Parameters

Parameter	Required	Data Type	Description
EventType	Yes	String	Unique name of the event type
EventId	Yes	String	Unique identifier of the event

## Request Example

```
/ES/EventType/core.automobiles:ExceededTemperature/v1/  
Events('F2A5A34034CE4B0DB2BCD10DD30F07C2')
```

Deletes the event with the ID F2A5A34034CE4B0DB2BCD10DD30F07C2 that belongs to the event type core.automobiles:ExceededTempeprature

## Response

### Response Status and Error Codes

Code	Description
204	Event deleted successfully

## Related Information

[User-defined Events based on OData Event Type \[page 998\]](#)

## 19.2.5 Read Metadata for Event

Retrieves metadata for events

With this service, you can retrieve metadata for events.

## Request

**URI:** /ES/v1/\$metadata

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <td>.r

## Request Parameters

None

## Request Example

/ES/v1/\$metadata

## Response

### Response Status and Error Codes

Code	Description
200	Metadata retrieved successfully

## Payload

Format: [XML](#)

### Example 1

/ES/v1/\$metadata

```
<EntityType Name="EventPropertySetTypes">
  <Key>
    <PropertyRef Name="Name"></PropertyRef>
  </Key>
  <Property Name="Name" Type="Edm.String" Nullable="false"></
Property>
  <Property Name="PackageName" Type="Edm.String"
Nullable="false"></Property>
  <Property Name="DataCategory" Type="Edm.String"></Property>
  <Property Name="Language" Type="Edm.String"></Property>
  <Property Name="PropertySetTypeDescription" Type="Edm.String"></
Property>
  <NavigationProperty Name="Properties"
Relationship="com.sap.appiot.es.EventPropertySetType_Property"
FromRole="EventPropertySetTypes" ToRole="Properties"></NavigationProperty>
  <NavigationProperty Name="Descriptions"
Relationship="com.sap.appiot.es.EventPropertySetType_Description"
FromRole="EventPropertySetTypes" ToRole="Descriptions"></NavigationProperty>
  <NavigationProperty Name="Annotations"
Relationship="com.sap.appiot.es.EventPropertySetType_Annotation"
FromRole="EventPropertySetTypes" ToRole="Annotations"></NavigationProperty>
</EntityType>
<EntityType Name="EventTypes">
  <Key>
    <PropertyRef Name="Name"></PropertyRef>
  </Key>
  <Property Name="Name" Type="Edm.String" Nullable="false"></
Property>
  <Property Name="PackageName" Type="Edm.String"
Nullable="false"></Property>
  <Property Name="PropertySetId" Type="Edm.String"></Property>
  <Property Name="PropertySetType" Type="Edm.String"></Property>
  <Property Name="LanguageCode" Type="Edm.String"></Property>
```

```

        <Property Name="Description" Type="Edm.String"></Property>
        <Property Name="EventTypeState" Type="Edm.String"></Property>
        <NavigationProperty Name="PropertySetDescriptions"
Relationship="com.sap.appiot.es.EventType_PropertySetDescription"
FromRole="EventTypes" ToRole="PropertySetDescriptions"></NavigationProperty>
        <NavigationProperty Name="Statuses"
Relationship="com.sap.appiot.es.EventType_Status" FromRole="EventTypes"
ToRole="Statuses"></NavigationProperty>
        <NavigationProperty Name="Severities"
Relationship="com.sap.appiot.es.EventType_Severity" FromRole="EventTypes"
ToRole="Severities"></NavigationProperty>
        <NavigationProperty Name="Codes"
Relationship="com.sap.appiot.es.EventType_Code" FromRole="EventTypes"
ToRole="Codes"></NavigationProperty>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.appiot.es.EventType_Description" FromRole="EventTypes"
ToRole="Descriptions"></NavigationProperty>
    </EntityType>
    <EntityType Name="Events">
        <Key>
            <PropertyRef Name="EventId"></PropertyRef>
        </Key>
        <Property Name="EventId" Type="Edm.String" Nullable="false">
            <Documentation>
                <LongDescription>Event unique ID</LongDescription>
            </Documentation>
        </Property>
        <Property Name="CorrelationId" Type="Edm.String">
            <Documentation>
                <LongDescription>Event Correlation ID</LongDescription>
            </Documentation>
        </Property>
        <Property Name="BusinessTimestamp" Type="Edm.String"></Property>
        <Property Name="Type" Type="Edm.String"></Property>
        <Property Name="EventType" Type="Edm.String"></Property>
        <Property Name="EventStatus" Type="Edm.String"></Property>
        <Property Name="EventSeverity" Type="Edm.Int64"></Property>
        <Property Name="EventCode" Type="Edm.String"></Property>
        <Property Name="EventSource" Type="Edm.String"></Property>
        <Property Name="ThingId" Type="Edm.String"></Property>
        <Property Name="ThingProperty" Type="Edm.String"></Property>
        <Property Name="ExternalId" Type="Edm.String"></Property>
        <Property Name="EventInfo" Type="Edm.String"></Property>
    </EntityType>
    <EntityType Name="Properties">
        <Key>
            <PropertyRef Name="Name"></PropertyRef>
            <PropertyRef Name="PropertySetType"></PropertyRef>
        </Key>
        <Property Name="Name" Type="Edm.String" Nullable="false"></
Property>
        <Property Name="PropertySetType" Type="Edm.String"
Nullable="false"></Property>
        <Property Name="Type" Type="Edm.String"></Property>
        <Property Name="PropertyLength" Type="Edm.String"></Property>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.appiot.es.Property_Description" FromRole="Properties"
ToRole="Descriptions"></NavigationProperty>
    </EntityType>
    <EntityType Name="Descriptions">
        <Key>
            <PropertyRef Name="LanguageCode"></PropertyRef>
        </Key>
        <Property Name="LanguageCode" Type="Edm.String">
            <Documentation>
                <LongDescription>Language Code</LongDescription>
            </Documentation>
        </Property>

```

```

        <Property Name="Description" Type="Edm.String">
            <Documentation>
                <LongDescription>Description</LongDescription>
            </Documentation>
        </Property>
    </EntityType>
    <EntityType Name="Annotations">
        <Key>
            <PropertyRef Name="Name"></PropertyRef>
        </Key>
        <Property Name="Name" Type="Edm.String" Nullable="false"
MaxLength="81" sap:creatable="true" sap:updatable="false">
            <Documentation>
                <LongDescription>Name of Annotation</LongDescription>
            </Documentation>
        </Property>
        <Property Name="PackageName" Type="Edm.String" Nullable="true"
MaxLength="50" sap:creatable="false" sap:updatable="false">
            <Documentation>
                <LongDescription>Package Name</LongDescription>
            </Documentation>
        </Property>
        <NavigationProperty Name="Annotations"
Relationship="com.sap.apptot.es.EventPropertySetType_Annotation"
FromRole="EventPropertySetTypes" ToRole="Annotations"></NavigationProperty>
    </EntityType>
    <EntityType Name="Codes">
        <Key>
            <PropertyRef Name="EventCode"></PropertyRef>
        </Key>
        <Property Name="EventCode" Type="Edm.String"></Property>
        <Property Name="LanguageCode" Type="Edm.String"></Property>
        <Property Name="Description" Type="Edm.String"></Property>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.es.Code_Description" FromRole="Codes"
ToRole="Descriptions"></NavigationProperty>
    </EntityType>
    <EntityType Name="Severity">
        <Key>
            <PropertyRef Name="EventSeverity"></PropertyRef>
        </Key>
        <Property Name="EventSeverity" Type="Edm.String"></Property>
        <Property Name="LanguageCode" Type="Edm.String"></Property>
        <Property Name="Description" Type="Edm.String"></Property>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.es.Severity_Description" FromRole="Severities"
ToRole="Descriptions"></NavigationProperty>
    </EntityType>
    <EntityType Name="PropertySetDescriptions">
        <Key>
            <PropertyRef Name="LanguageCode"></PropertyRef>
        </Key>
        <Property Name="LanguageCode" Type="Edm.String" Nullable="false">
            <Documentation>
                <LongDescription>Language Code</LongDescription>
            </Documentation>
        </Property>
        <Property Name="Description" Type="Edm.String">
            <Documentation>
                <LongDescription>Description</LongDescription>
            </Documentation>
        </Property>
    </EntityType>
    <EntityType Name="EventCorrelations">
        <Key>
            <PropertyRef Name="CorrelationId"></PropertyRef>
        </Key>
        <Property Name="CorrelationId" Type="Edm.String">

```

```

        <Documentation>
            <LongDescription>Event Correlation ID</LongDescription>
        </Documentation>
    </Property>
    <Property Name="EventId" Type="Edm.String"></Property>
    <Property Name="BusinessTimestamp" Type="Edm.String"></Property>
    <NavigationProperty Name="Events"
Relationship="com.sap.appiot.es.EventCorrelation_Event"
FromRole="EventCorrelations" ToRole="Events"></NavigationProperty>
    </EntityType>
    <EntityType Name="Status">
        <Key>
            <PropertyRef Name="EventStatus"></PropertyRef>
        </Key>
        <Property Name="EventStatus" Type="Edm.String"></Property>
        <Property Name="LanguageCode" Type="Edm.String"></Property>
        <Property Name="Description" Type="Edm.String"></Property>
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.appiot.es.Status_Description" FromRole="Statuses"
ToRole="Descriptions"></NavigationProperty>
    </EntityType>
    <Association Name="EventType_PropertySetDescription">
        <End Type="com.sap.appiot.es.EventType" Multiplicity="1"
Role="EventTypes"></End>
        <End Type="com.sap.appiot.es.PropertySetDescription"
Multiplicity="*" Role="PropertySetDescriptions"></End>
    </Association>
    <Association Name="EventType_Code">
        <End Type="com.sap.appiot.es.EventType" Multiplicity="1"
Role="EventTypes"></End>
        <End Type="com.sap.appiot.es.Code" Multiplicity="*"
Role="Codes"></End>
    </Association>
    <Association Name="EventCorrelation_Event">
        <End Type="com.sap.appiot.es.EventCorrelation" Multiplicity="1"
Role="EventCorrelations"></End>
        <End Type="com.sap.appiot.es.Event" Multiplicity="*"
Role="Events"></End>
    </Association>
    <Association Name="EventType_Description">
        <End Type="com.sap.appiot.es.EventType" Multiplicity="1"
Role="EventTypes"></End>
        <End Type="com.sap.appiot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>
    <Association Name="Severity_Description">
        <End Type="com.sap.appiot.es.Severity" Multiplicity="1"
Role="Severities"></End>
        <End Type="com.sap.appiot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>
    <Association Name="Property_Description">
        <End Type="com.sap.appiot.es.Property" Multiplicity="1"
Role="Properties"></End>
        <End Type="com.sap.appiot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>
    <Association Name="Status_Description">
        <End Type="com.sap.appiot.es.Status" Multiplicity="1"
Role="Statuses"></End>
        <End Type="com.sap.appiot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>
    <Association Name="EventPropertySetType_Description">
        <End Type="com.sap.appiot.es.EventPropertySetType"
Multiplicity="1" Role="EventPropertySetTypes"></End>
        <End Type="com.sap.appiot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
    </Association>

```

```

        </Association>
        <Association Name="EventType_Status">
            <End Type="com.sap.appiot.es.EventType" Multiplicity="1"
Role="EventTypes"></End>
            <End Type="com.sap.appiot.es.Status" Multiplicity="*"
Role="Statuses"></End>
        </Association>
        <Association Name="EventType_Severity">
            <End Type="com.sap.appiot.es.EventType" Multiplicity="1"
Role="EventTypes"></End>
            <End Type="com.sap.appiot.es.Severity" Multiplicity="*"
Role="Severities"></End>
        </Association>
        <Association Name="Code_Description">
            <End Type="com.sap.appiot.es.Code" Multiplicity="1"
Role="Codes"></End>
            <End Type="com.sap.appiot.es.Description" Multiplicity="*"
Role="Descriptions"></End>
        </Association>
        <Association Name="EventPropertySetType_Annotation">
            <End Type="com.sap.appiot.es.EventPropertySetType"
Multiplicity="1" Role="EventPropertySetTypes"></End>
            <End Type="com.sap.appiot.es.Annotations" Multiplicity="*"
Role="Annotations"></End>
        </Association>
        <Association Name="EventPropertySetType_Property">
            <End Type="com.sap.appiot.es.EventPropertySetType"
Multiplicity="1" Role="EventPropertySetTypes"></End>
            <End Type="com.sap.appiot.es.Property" Multiplicity="*"
Role="Properties"></End>
        </Association>
        <EntityContainer Name="UnifiedEvents"
m:IsDefaultEntityContainer="true">
            <EntitySet Name="EventPropertySetTypes"
EntityType="com.sap.appiot.es.EventPropertySetType"></EntitySet>
            <EntitySet Name="EventTypes"
EntityType="com.sap.appiot.es.EventType"></EntitySet>
            <EntitySet Name="Events" EntityType="com.sap.appiot.es.Event"></
EntitySet>
            <EntitySet Name="Properties"
EntityType="com.sap.appiot.es.Property"></EntitySet>
            <EntitySet Name="Descriptions"
EntityType="com.sap.appiot.es.Description"></EntitySet>
            <EntitySet Name="Annotations"
EntityType="com.sap.appiot.es.Annotations"></EntitySet>
            <EntitySet Name="Codes" EntityType="com.sap.appiot.es.Code"></
EntitySet>
            <EntitySet Name="Severities"
EntityType="com.sap.appiot.es.Severity"></EntitySet>
            <EntitySet Name="PropertySetDescriptions"
EntityType="com.sap.appiot.es.PropertySetDescription"></EntitySet>
            <EntitySet Name="EventCorrelations"
EntityType="com.sap.appiot.es.EventCorrelation"></EntitySet>
            <EntitySet Name="Statuses"
EntityType="com.sap.appiot.es.Status"></EntitySet>
            <AssociationSet Name="EventType_PropertySetDescription"
Association="com.sap.appiot.es.EventType_PropertySetDescription">
                <End EntitySet="EventTypes" Role="EventTypes"></End>
                <End EntitySet="PropertySetDescriptions"
Role="PropertySetDescriptions"></End>
            </AssociationSet>
            <AssociationSet Name="EventType_Code"
Association="com.sap.appiot.es.EventType_Code">
                <End EntitySet="EventTypes" Role="EventTypes"></End>
                <End EntitySet="Codes" Role="Codes"></End>
            </AssociationSet>
            <AssociationSet Name="EventCorrelation_Event"
Association="com.sap.appiot.es.EventCorrelation_Event">

```

```

        <End EntitySet="EventCorrelations"
Role="EventCorrelations"></End>
        <End EntitySet="Events" Role="Events"></End>
    </AssociationSet>
    <AssociationSet Name="EventType_Description"
Association="com.sap.appiot.es.EventType_Description">
        <End EntitySet="EventTypes" Role="EventTypes"></End>
        <End EntitySet="Descriptions" Role="Descriptions"></End>
    </AssociationSet>
    <AssociationSet Name="Severity_Description"
Association="com.sap.appiot.es.Severity_Description">
        <End EntitySet="Severities" Role="Severities"></End>
        <End EntitySet="Descriptions" Role="Descriptions"></End>
    </AssociationSet>
    <AssociationSet Name="Property_Description"
Association="com.sap.appiot.es.Property_Description">
        <End EntitySet="Properties" Role="Properties"></End>
        <End EntitySet="Descriptions" Role="Descriptions"></End>
    </AssociationSet>
    <AssociationSet Name="Status_Description"
Association="com.sap.appiot.es.Status_Description">
        <End EntitySet="Statuses" Role="Statuses"></End>
        <End EntitySet="Descriptions" Role="Descriptions"></End>
    </AssociationSet>
    <AssociationSet Name="EventPropertySetType_Description"
Association="com.sap.appiot.es.EventPropertySetType_Description">
        <End EntitySet="EventPropertySetTypes"
Role="EventPropertySetTypes"></End>
        <End EntitySet="Descriptions" Role="Descriptions"></End>
    </AssociationSet>
    <AssociationSet Name="EventType_Status"
Association="com.sap.appiot.es.EventType_Status">
        <End EntitySet="EventTypes" Role="EventTypes"></End>
        <End EntitySet="Statuses" Role="Statuses"></End>
    </AssociationSet>
    <AssociationSet Name="EventType_Severity"
Association="com.sap.appiot.es.EventType_Severity">
        <End EntitySet="EventTypes" Role="EventTypes"></End>
        <End EntitySet="Severities" Role="Severities"></End>
    </AssociationSet>
    <AssociationSet Name="Code_Description"
Association="com.sap.appiot.es.Code_Description">
        <End EntitySet="Codes" Role="Codes"></End>
        <End EntitySet="Descriptions" Role="Descriptions"></End>
    </AssociationSet>
    <AssociationSet Name="EventPropertySetType_Annotation"
Association="com.sap.appiot.es.EventPropertySetType_Annotation">
        <End EntitySet="EventPropertySetTypes"
Role="EventPropertySetTypes"></End>
        <End EntitySet="Annotations" Role="Annotations"></End>
    </AssociationSet>
    <AssociationSet Name="EventPropertySetType_Property"
Association="com.sap.appiot.es.EventPropertySetType_Property">
        <End EntitySet="EventPropertySetTypes"
Role="EventPropertySetTypes"></End>
        <End EntitySet="Properties" Role="Properties"></End>
    </AssociationSet>
</EntityContainer>

```

## Related Information

[User-defined Events based on OData Event Type \[page 998\]](#)



## 19.2.6 Read Events

With this method you can retrieve all events.

The service retrieves static data for all events that belong to user-defined event type and standard event type with the property set type data.

In addition, you can retrieve a subset of all events, according to the filter criteria provided. For a collective request for a set of events, you have these options:

- You can restrict the set of events to be retrieved by filter criteria based on all fields available in the response.
- You can define that the set of matching events shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 20 events per response. You can navigate to the next set of results, if available, using the **\_\_next** link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

### Request

**URI:** `/ES/v1/Events`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

### Request Parameters

None

### Query Request Parameters

Parameter	Type	Required	Description
<code>\$top</code>	Integer	No	Number of records to include in the result set.
<code>\$skip</code>	Integer	No	Number of records to exclude from the result set.

Parameter	Type	Required	Description
\$orderby	String	No	Field name to be used for sorting the result set in ascending order; valid fields are <i>CorrelationId</i> , <i>BusinessTimestamp</i> , <i>EventType</i> , <i>EventStatus</i> , <i>EventSeverity</i> , <i>EventCode</i> , <i>ThingId</i> , <i>ThingProperty</i> , and <i>ExternalId</i> .  To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.
\$select	String	No	Select condition for the data fields to be included in the result set; valid fields are <i>CorrelationId</i> , <i>BusinessTimestamp</i> , <i>EventType</i> , <i>EventStatus</i> , <i>EventSeverity</i> , <i>EventCode</i> , <i>ThingId</i> , <i>ThingProperty</i> , and <i>ExternalId</i> .
\$filter	String	No	Filter condition to be applied; all fields in the result set are valid.

## Request Example

/ES/v1/Events

Retrieves static data for all events that belong to user-defined event type and standard event type with the property set type data

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: *JSON*

Media types *JSON* and *XML* are supported. The *JSON* for this method has the structure as depicted in the following example:

### Example 1

/ES/v1/Events

Retrieves static data for all events that belong to user-defined event type and standard event type with the property set type data

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('9DDBBD81FD904DBD8A276AA25899F5CE')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('9DDBBD81FD904DBD8A276AA25899F5CE')",
        "type": "com.sap.apptot.es.Events"
      },
      "EventId": "F392018DE4D54CCFB5FEC72672D4310B",
      "CorrelationId": "BD622E3DA6A14C4D87CDEC935B000001",
      "BusinessTimestamp": "2017-04-08T12:30:00Z",
      "Type": "Alert",
      "EventType": "core.automobiles:AlertEventType",
      "EventInfo": "Alert on temperature",
      "EventStatus": "going",
      "EventSeverity": "2",
      "EventCode": "EventCode2",
      "EventSource": "Test Source01304",
      "ThingId": "0BCDD3EB71DB4935BAF9603577BF73F4",
      "ThingProperty": "core.automobiles:AlertEventPST/alertEvent",
      "ExternalId": "AlertExternal"
    },
    {
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('3DAD618BA9354232B90FC79A5B542DE6')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('3DAD618BA9354232B90FC79A5B542DE6')",
        "type": "com.sap.apptot.es.Events"
      },
      "EventId": "5DDF968BDF12404E8D8E1FA301427209",
      "CorrelationId": "3DAD618BA9354232B90FC79A5B542DE6",
      "BusinessTimestamp": "2017-03-12T05:30:00Z",
      "Type": "Important",
      "EventType": "com.sap.apptot.eventtypes:StandardEventType",
      "EventInfo": "Alert on temperature",
      "EventStatus": "Open",
      "EventSeverity": "2",
      "EventCode": "30",
      "EventSource": "Car urgent event",
      "ThingId": "155ABA08434C4F95B56C0A84D8D30ABE",
      "ThingProperty": "core.automobiles:Car/wheel_fl/Temperature",
      "ExternalId": "CI_Pressure-Urgent-PU4312"
    },
    {
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('12644D8B35624BF7A85B1A425B26E700')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('12644D8B35624BF7A85B1A425B26E700')",
        "type": "com.sap.apptot.es.Events"
      },
      "EventId": "12644D8B35624BF7A85B1A425B26E700",
      "CorrelationId": "3DAD618BA9354232B90FC79A5B542DE6",
      "BusinessTimestamp": "2017-04-19T12:00:00Z",
```

```

        "Type": "com.sap.apptot.eventtypes:StandardEventType",
        "EventType": "com.sap.apptot.eventtypes:StandardEventType",
        "EventInfo": "Alert on temperature",
        "EventStatus": "Open",
        "EventSeverity": "3",
        "EventCode": "30",
        "EventSource": "Drilling hole length measurer",
        "ThingId": "237FC64161FC46C2A72AB8BE3533C1B0",
        "ThingProperty": "core.automobiles:dd350/dd350_holes/
Hole_Info_Length",
        "ExternalId": "XW5619"
    },
    {
        "__metadata": {
            "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/Events('50F70DDB6CD44B77ACBA61A3DE09B7D2')",
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/Events('50F70DDB6CD44B77ACBA61A3DE09B7D2')",
            "type": "com.sap.apptot.es.Events"
        },
        "EventId": "50F70DDB6CD44B77ACBA61A3DE09B7D2",
        "CorrelationId": "3DAD618BA9354232B90FC79A5B542DE6",
        "BusinessTimestamp": "2016-01-01T09:02:00Z",
        "Type": "Warning",
        "EventType": "com.sap.apptot.eventtypes:StandardEventType",
        "EventInfo": "Alert on temperature",
        "EventStatus": "Completed",
        "EventSeverity": "1",
        "EventCode": "30",
        "EventSource": "Generator",
        "ThingId": "483BA41481A4450D99010F4CFDE3B9A6",
        "ThingProperty": "core.automobiles:ABCXSeries/wheel_fr",
        "ExternalId": "eevents3_reg110"
    },
    {
        "__metadata": {
            "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/Events('EDCB5EF081C2439AB658B231631B1F7F')",
            "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/Events('EDCB5EF081C2439AB658B231631B1F7F')",
            "type": "com.sap.apptot.es.Events"
        },
        "EventId": "EDCB5EF081C2439AB658B231631B1F7F",
        "CorrelationId": "c39d9066076840e18c3718ee77b4c353",
        "BusinessTimestamp": "2017-03-24T06:00:00Z",
        "Type": "Alert",
        "EventType": "core.automobiles:SampleEventType",
        "EventInfo": "Alert on temperature",
        "EventStatus": "open",
        "EventSeverity": "2",
        "EventCode": "EventCode1",
        "EventSource": "Test Source13687",
        "ThingId": "AB6113709D044F208C77DF2D23B42153",
        "ThingProperty": "core.automobiles:SampleEventPST/sampleEvent",
        "ExternalId": "Perf_TypedEvents_13687"
    }
],
    "__next": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/
Events?$skiptoken=20"
}
}

```

## Related Information

[User-defined Events based on OData Event Type \[page 998\]](#)

### 19.2.7 Read Event Correlations

With this method, you can retrieve a unique list of correlation IDs and the latest event corresponding to each correlation ID.

In addition, you can retrieve a subset of all correlation IDs, according to the filter criteria provided. For a collective request for a set of correlation IDs, you have these options:

- You can restrict the set of correlation IDs to be retrieved by filter criteria based on the correlation ID, business timestamp, and event ID.
- You can define that the set of matching correlation IDs shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/ES/v1/EventCorrelations`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

## Request Example

`/ES/v1/EventCorrelations`

## Query Request Parameter

Parameter	Type	Required	Description
<code>\$top</code>	Integer	No	Number of records to include in the result set.
<code>\$skip</code>	Integer	No	Number of records to exclude from the result set.

Parameter	Type	Required	Description
\$orderby	String	No	Field name to be used for sorting the result set in ascending order; valid fields are <a href="#">CorrelationId</a> , <a href="#">BusinessTimestamp</a> , and <a href="#">EventId</a> .  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$select	String	No	Select condition for the data fields to be included in the result set; valid fields are <a href="#">CorrelationId</a> , <a href="#">BusinessTimestamp</a> , and <a href="#">EventId</a> .
\$filter	String	No	Filter condition to be applied; valid fields are <a href="#">CorrelationId</a> , <a href="#">BusinessTimestamp</a> , and <a href="#">EventId</a> .

## Response

### Response Status and Error Codes

Code	Description
200	List of events retrieved successfully

## Payload

**Format:** [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as illustrated in the following example.

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventCorrelations('BD622E3DA6A14C4D87CDEC935B000001')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/EventCorrelations('BD622E3DA6A14C4D87CDEC935B000001')",
        "type": "com.sap.apptot.es.EventCorrelations"
      },
      ...
    ]
  }
}
```

```

        "CorrelationId": "BD622E3DA6A14C4D87CDEC935B000001",
        "EventId": "F392018DE4D54CCFB5FEC72672D4310B",
        "BusinessTimestamp": "2017-02-14T06:30:00Z",
        "Events": {
            "__deferred": {
                "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/
EventCorrelations('BD622E3DA6A14C4D87CDEC935B000001')/Events"
            }
        },
        {
            "__metadata": {
                "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventCorrelations('c39d9066076840e18c3718ee77b4c353')",
                "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventCorrelations('c39d9066076840e18c3718ee77b4c353')",
                "type": "com.sap.apptot.es.EventCorrelations"
            },
            "CorrelationId": "c39d9066076840e18c3718ee77b4c353",
            "EventId": "3139A778A683470EBD00BCC4B0543B40",
            "BusinessTimestamp": "2017-02-14T07:00:00Z",
            "Events": {
                "__deferred": {
                    "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/
EventCorrelations('c39d9066076840e18c3718ee77b4c353')/Events"
                }
            }
        },
        {
            "__metadata": {
                "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventCorrelations('5b9954a4f0a74c6f96b10e744395cac6')",
                "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:
443/ES/v1/EventCorrelations('5b9954a4f0a74c6f96b10e744395cac6')",
                "type": "com.sap.apptot.es.EventCorrelations"
            },
            "CorrelationId": "5b9954a4f0a74c6f96b10e744395cac6",
            "EventId": "2C7240C29D724D5ABC991E0054243A39",
            "BusinessTimestamp": "2017-02-15T03:00:00Z",
            "Events": {
                "__deferred": {
                    "uri": "https://events-sap-
ci.cfapps.sap.hana.ondemand.com:443/ES/v1/
EventCorrelations('5b9954a4f0a74c6f96b10e744395cac6')/Events"
                }
            }
        }
    ]
}

```

## Related Information

[User-defined Events based on OData Event Type \[page 998\]](#)

## 19.2.8 Read Events with a Correlation ID

With this service, you can retrieve all events with the specified correlation ID. All events with the same correlation ID belongs to a specific event type and they describe the life cycle of an event.

By default, the service displays a maximum of 500 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

### Request

**URI:** `/ES/v1/EventCorrelations('correlation ID')/Events`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
<code>correlationId</code>	Yes	String	Unique identifier to group all related events	Path

### Request Example

`/ES/v1/EventCorrelations('c39d9066076840e18c3718ee77b4c353')/Events`

### Query Request Parameter

Parameter	Type	Required	Description
<code>\$top</code>	Integer	No	Number of records to include in the result set. Default value is 20.
<code>\$skip</code>	Integer	No	Number of records to exclude from the result set. Default value is 0.



Parameter	Type	Required	Description
\$orderby	String	No	Field name to be used for sorting the result set in ascending order; valid fields are <i>CorrelationId</i> , <i>BusinessTimestamp</i> , <i>EventType</i> , <i>EventStatus</i> , <i>EventSeverity</i> , <i>EventCode</i> , <i>ThingId</i> , <i>ThingProperty</i> , and <i>ExternalId</i> .  To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.
\$select	String	No	Select condition for the data fields to be included in the result set; valid fields are <i>CorrelationId</i> , <i>BusinessTimestamp</i> , <i>EventType</i> , <i>EventStatus</i> , <i>EventSeverity</i> , <i>EventCode</i> , <i>ThingId</i> , <i>ThingProperty</i> , and <i>ExternalId</i> .
\$filter	String	No	Filter condition to be applied; valid fields are <i>CorrelationId</i> , <i>EventType</i> , <i>EventStatus</i> , <i>EventSeverity</i> , <i>EventCode</i> , <i>ThingId</i> , <i>ThingProperty</i> , <i>BusinessTimestamp</i> , and <i>ExternalId</i> .

## Response

### Response Status and Error Codes

Code	Description
200	List of events with specific correlation ID retrieved successfully

## Payload

Format: *JSON*

Media types *JSON* and *XML* are supported. The *JSON* for this method has the structure as depicted in the following example.

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('5DDF968BDF12404E8D8E1FA301427209')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('5DDF968BDF12404E8D8E1FA301427209')",
        "type": "com.sap.apptot.es.Events"
      },
      "EventId": "5DDF968BDF12404E8D8E1FA301427209",
      "CorrelationId": "c39d9066076840e18c3718ee77b4c353",
      "BusinessTimestamp": "2017-11-02T02:40:00Z",
      "Type": "Alert",
      "EventType": "core.bbautomobiles:ExceededTemperature",
      "EventStatus": "Open",
      "EventSeverity": "1",
      "EventCode": "EQ12",
      "EventSource": null,
      "ThingId": "D6B0B87A025B473C94E2F9421A1277C8",
      "ThingProperty": null,
      "ExternalId": "SampleEvent",
      "EventInfo": ""
    },
    {
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('9C9DFBCB4A2C4F76A953A459355AEE0B')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('9C9DFBCB4A2C4F76A953A459355AEE0B')",
        "type": "com.sap.apptot.es.Events"
      },
      "EventId": "9C9DFBCB4A2C4F76A953A459355AEE0B",
      "CorrelationId": "c39d9066076840e18c3718ee77b4c353",
      "BusinessTimestamp": "2017-11-02T02:42:00Z",
      "Type": "Alert",
      "EventType": "core.bbautomobiles:ExceededTemperature",
      "EventStatus": "InProgress",
      "EventSeverity": "1",
      "EventCode": "EQ12",
      "EventSource": null,
      "ThingId": "D6B0B87A025B473C94E2F9421A1277C8",
      "ThingProperty": null,
      "ExternalId": "SampleEvent",
      "EventInfo": ""
    },
    {
      "__metadata": {
        "id": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('ABC88B4C16344E4DAEDAD6F7C01C88FE')",
        "uri": "https://events-sap-ci.cfapps.sap.hana.ondemand.com:443/ES/v1/Events('ABC88B4C16344E4DAEDAD6F7C01C88FE')",
        "type": "com.sap.apptot.es.Events"
      },
      "EventId": "ABC88B4C16344E4DAEDAD6F7C01C88FE",
      "CorrelationId": "c39d9066076840e18c3718ee77b4c353",
      "BusinessTimestamp": "2017-11-02T09:14:00Z",
      "Type": "Alert",
      "EventType": "core.bbautomobiles:ExceededTemperature",
      "EventStatus": "Completed",
      "EventSeverity": "1",
      "EventCode": "EQ12",
      "EventSource": null,
      "ThingId": "D6B0B87A025B473C94E2F9421A1277C8",
```

```

        "ThingProperty": null,
        "ExternalId": "SampleEvent",
        "EventInfo": ""
      }
    ]
  }
}

```

## Related Information

[User-defined Events based on OData Event Type \[page 998\]](#)

## 19.2.9 Delete an Event Using Correlation ID

With this method, you can delete one or more events with the specific correlation ID. The corresponding event data is also deleted. This method only supports only those events of event type created using the OData service (/ES/v1/EventTypes).

### Note

On deletion of the event, the system generates audit logs. If you have assigned one of the predefined annotations (`com.sap.appiot.security:pii` or `com.sap.appiot.security:spi`) for the property set type of an event type.

## Request

**URI:** /ES/v1/EventCorrelations('<correlation ID>')

**Operation Type:** CRUD

**HTTP Method:** *DELETE*

**Permissions:** <td>.d

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
correlationId	Yes	String	Correlation ID of the event	Path

### Request Example

/ES/v1/EventCorrelations('F2A5A34034CE4B0DB2BCD10DD30F07C2')

Deletes the event with the correlation ID F2A5A34034CE4B0DB2BCD10DD30F07C2 that belongs to the event type `core.automobiles:ExceededTemperature`

## Response

### Response Status and Error Codes

Code	Description
204	Event deleted successfully

### Related Information

[User-defined Events based on OData Event Type \[page 998\]](#)

# 20 File

The File services allow you to store files or documents for a thing.

## Base URI

**Formal description:** `http://<server address>[:<port number>][/path]/Files`

**Example for a base URI in a cloud foundry environment:** `https://appiot-fs.cfapps.eu10.hana.ondemand.com/Files`

### Note

App name is `<appiot-fs>`.

You cannot use the following special characters in the filename or the path name:

Ampersand (&)	Dollar (\$)	'At' symbol (@)	Equals (=)
Colon (:)	Plus (+)	Semicolon (;)	Comma (,)
Question mark (?)	Backslash (\)	Left curly brace { }	Right curly brace { }
Caret (^)	Percent character (%)	Grave accent / back tick (`)	Left square bracket ([)
Right square bracket (])	'Greater Than' symbol (>)	'Less Than' symbol (<)	'Pound' character (#)
Tilde (~)	Vertical bar / pipe ( )	Double quotes (")	Double forward slash (//)

## Methods

HTTP Method	Action	URI	Scopes
<i>PUT</i>	<a href="#">Upload a File [page 1034]</a>	<code>/Files/Things/&lt;thing id&gt;/&lt;path&gt;/&lt;filename&gt;</code>	<code>&lt;File&gt;.c</code>
<i>GET</i>	<a href="#">Read a File [page 1036]</a>	<code>/Files/Things/&lt;thing id&gt;/&lt;path&gt;/&lt;filename&gt;</code>	<code>&lt;File&gt;.r</code>
<i>GET</i>	<a href="#">Read Metadata [page 1038]</a>	<code>/Files/Things/&lt;thing id&gt;/&lt;path&gt;/</code>	<code>&lt;File&gt;.r</code>
<i>DELETE</i>	<a href="#">Delete a File [page 1039]</a>	<code>/Files/Things/&lt;thing id&gt;/&lt;path&gt;/&lt;filename&gt;</code>	<code>&lt;File&gt;.d</code>

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read Version Information of File Service [page 1041]</a>	/Version	<File>.r

### Note

<File> refers to the file application name.

## 20.1 Upload a File

This file service allows you to upload a file in a folder or a particular path.

When you upload a file, the service scans the file using the **Malware Scanner** service provided by Cloud Foundry. After scanning, the scanner service alerts the user for the presence of malware, if any.

- If the scanning service fails to scan the file, the file service continues to upload the file with a warning message about the failed scan. The warning message is available in the response header. The files that are not scanned are not retrieved using the GET call.
- If the scanning succeeds, the file service uploads the file only if it is malware free and if it belongs to one of the following mime-types:

application/msexcel	application/	audio/wav
application/mspowerpoint	vnd.openxmlformats-	image/gif
application/msword	officedocument.wordprocess	image/jpeg
application/octet-stream	ingml.document	image/png
application/pdf	application/	image/png
application/plain	vnd.openxmlformats-	text/plain
application/rtf	officedocument.spreadsheet	text/richtext
application/vnd.ms-outlook	ml.sheet	text/richtext
	application/	video/avi
	vnd.openxmlformats-	video/mpeg
	officedocument.presentation	
	nml.presentation	
	application/zip	
	audio/mpeg	
	audio/midi	

### Note

- If the file is of a mime-type that is not supported, the file upload operation is terminated.
- If you need to upload a file of mime-type that is not supported, please raise a support ticket under the component IOT-BSV-TS.

You can assign the predefined annotations `com.sap.appiot.security:pii` and `com.sap.appiot.security:spi` while creating a file. As a result, the content of the file is considered as personal data or sensitive data based on the annotation defined. For such files, whenever the content is retrieved, the content is created or updated, or the content is deleted, the system generates audit logs. After creating the file with predefined annotation, you cannot remove or change the annotation.

#### **i** Note

- Filename is mandatory.
- Path is optional.
- URL must not end with '/'. A URL that ends with a '/' is considered a folder.
- You can only upload a file of size less than or equal to 20MB.

## Request

**URI:** `/Files/Things/<thing id>/<path>/<filename>`

**HTTP Method:** `PUT` or `POST`

#### **i** Note

You can create a file using the `PUT` or the `POST` method. We recommend to use the `POST` method to create a file.

#### Permissions:

`<File>.c`

## Request Parameter

Parameter	Required	Data Type	Description
id	Yes	string	Internal identifier for a thing.
path	No	string	Path where the file exists.
name	Yes	string	Name of the file.

## Request Headers

Header	Required	Values
Content-Type	Yes	binary/octet-stream
sap-iot-audit-log	No	You can assign one of the following values: <ul style="list-style-type: none"><li>• <code>com.sap.appiot.security:pii</code></li><li>• <code>com.sap.appiot.security:spi</code></li></ul>

## Request Example

- File upload request with folder details

```
/Files/Things/0023A6329DE5491699ADF9E00EF431FB/myfolder1/myfolder2/myfile.txt
```

- File upload request without folder details

```
/Files/Things/0023A6329DE5491699ADF9E00EF431FB/myfile.txt
```

## Response

Format: *JSON*

### Response Status and Error Codes

Code	Description
201	File uploaded successfully

## 20.2 Read a File

This file service is used to read the content of a file.

If the file was scanned seven days ago or not scanned earlier, the system triggers the scanning. As a result, if the file contains malware or does not belong to the supported mime-type, the system deletes the file and displays an appropriate error message.

### Note

- Filename is mandatory. You must provide the same extension used during the uploading of the file.
- Path is optional.
- URL must not end with '/'. A URL that ends with a '/' is considered a folder.

## Request

**URI:** /Files/Things/<thing id>/<path>/<filename>

**HTTP Method:** *GET*

**Permissions:**

<File>.r



## Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header <code>CONTENT-ENCODING</code> with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.
Cache-Control: max-age	No	Time duration in seconds  Indicates the maximum amount of time in seconds for which the fetched responses can be reused (from the time the request is triggered). For example, <code>max-age=120</code> indicates that fetched response can be reused (remains in the browser cache) for the next 120 seconds.

## Request Parameter

Parameter	Required	Data Type	Description
id	Yes	string	Internal identifier for a thing.
path	No	string	Path where the file exists.
name	Yes	string	Name of the file.

## Request Example

- Read the content of a file from a folder

```
/Files/Things/123/myfolder1/myfolder2/myfile.txt
```

- Read a file

```
/Files/Things/123/myfile.txt
```

## Response

Format: [JSON](#)

### Response Status and Error Codes

Code	Description
200	File retrieved successfully

### Response Example

```
My First File
```

## 20.3 Read Metadata

This service is used to read the metadata of all files and folders within a folder or within a specified path.

#### Note

- Path is optional.
- URL must end with '/'. A URL without a '/' is considered a request to read the file content.

### Request

**URI:** /Files/Things/<thing id>/<path>/

**HTTP Method:** [GET](#)

**Permissions:**

<File>.r

### Request Headers

Parameter	Required	Description
Accept-Encoding	No	Retrieve compressed response  Only if you specify the value <a href="#">gzip</a> or <a href="#">gzip,deflate</a> for this parameter, the system returns the compressed response with an additional response header CONTENT-ENCODING with value <a href="#">gzip</a> . If you specify an invalid value or do not use this header parameter, the system returns the uncompressed response.

## Request Parameter

Parameter	Required	Data Type	Description
id	Yes	string	Internal identifier for a thing.
path	No	string	Path where the file exists.

## Request Example

```
/Files/Things/123/myfolder1/myfolder2/?$top=10
```

## Response

Format: *JSON*

## Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: *JSON*

Only media type *JSON* is supported. The *JSON* for this method has the following structure:

```
[{
  "name": "myfile.txt",
  "length": 13,
  "mimeType": "binary/octet-stream",
  "user": "allpermission01@gmail.com",
  "isDirectory": false,
  "lastUpdateTime": "2016-02-26T04:42:00.743Z"
}, {
  "name": "myfolder2",
  "isDirectory": true
}]
```

## 20.4 Delete a File

This file service is used to delete a file.

### Note

- Filename is mandatory.

- Path is optional.
- You can only delete one file at a given point in time. You cannot delete a folder, and the URL must not end with a '/'.

## Request

**URI:** /Files/Things/<Thing ID>/<Path>/<Filename>

**HTTP Method:** *DELETE*

**Permissions:**

<File>.d

## Request Parameter

Parameter	Required	Data Type	Description
id	Yes	string	Internal identifier for a thing.
path	No	string	Path where the file exists.
name	Yes	string	Name of the file.

## Request Example

- Delete a file from a specific folder

```
/Files/Things/123/myfolder1/myfolder2/myfile.txt
```

- Delete a file

```
/Files/Things/123/myfile.txt
```

## Response

**Format:** *JSON*

### Response Status and Error Codes

Code	Description
204	File deleted successfully

## 20.5 Read Version Information of File Service

This service is used to retrieve the version information of the file service.

### Request

URI: `/Version`

HTTP Method: `GET`

Permissions:

`<File>.r`

### Request Parameters

None

### Request Example

```
/Version
```

### Response

Format: `JSON`

### Response Status and Error Codes

Code	Description
200	Version information retrieved successfully.

### Response Example

```
{  
  "version" : "1.10"  
}
```

## 20.6 HTTP Status Codes for File Services

List of status codes for file services

In the following table, you find the list of HTTP status codes that are used by the file services:

Put	Delete	Get	Description
201	204	200	Success
404	404	404	Capabilities not assigned to read a file that belongs to a specific thing.
201	204	404	Read capability not assigned to read a file that belongs to a specific thing. However, the user is assigned with the capabilities to perform operations such as create, update, and delete.
404	404	200	Read capability assigned to read a file that belongs to a specific thing. However, the user is not assigned with capabilities to perform operations such as create, update, and delete.
201	404	404	Not found. The requested file is not found.
403	403	403	Forbidden (scope(s) not assigned for the user)
400	400	400	Bad request (for example: header field missing, validation failed)
500	500	500	Internal server error

# 21 Composites

The composite services are used to retrieve a combination of data by joining multiple tables and filter, select, page, and sort the retrieved data according to the criteria provided.

## Related Information

[Composite Things: REST Service \[page 1043\]](#)

[Composite Things: OData Service \[page 1047\]](#)

[Composite Events: OData Service \[page 1062\]](#)

[Composite Thing Configuration for a Specific Thing Hierarchy: OData Service \[page 1093\]](#)

[Thing Detail \[page 1121\]](#)

[Advanced Thing List \[page 1136\]](#)

## 21.1 Composite Things: REST Service

You can use a thing service (GET <Composite Things Application URL>/Composite/Things/PropertySet/propertySetType name) to return the things with the most recent property data (such as time series) for the specified property set type. However, this service does not retrieve data about things, customers, and events in a single request.

### Note

The composite service only retrieves data for property set type of type *MasterData*.

**Base URI:** `http://<server address>[:<port number>][/path]/Composite/Things/PropertySet`

**Example for a base URI in a cloud foundry environment:** `https://composite-things.cfapps.eu10.hana.ondemand.com/Composite/Things/PropertySet`

## Methods

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read Thing Details with Property Set Type Data [page 1044]</a>	/Composite/Things/PropertySet/<propertySetType name>/<property set>	<ct>.r

HTTP Method	Action	URI	Scopes
GET	<a href="#">Read Version Information of Composite Service [page 1046]</a>	/Version	<ct>.r

#### i Note

<ct> refers to the composite application name.

## 21.1.1 Read Thing Details with Property Set Type Data

This service is used to retrieve the complete information of a thing for a specific property set type and a property set. The retrieved information includes the basic details (ID, external ID, thing type, and object group), customer details (customer name and contact), location details (address, country, longitude, and latitude), and event details (event count and event severity count).

#### i Note

The result set does not contain the fields longitude and latitude as part of thing location details, –

- If you did not specify values for these fields while creating a **Location**.
- If you did not specify the location of a thing while creating a **Thing**.

In addition, this service retrieves information about the usage of the properties and its latest value.

#### i Note

Retrieving number of objects more than the recommended limits with only one collective GET request can lead to high system load and a significant increase of response times.

To avoid performance decline, the system in future versions will automatically limit collective GET requests to a maximum of 100 objects that can be retrieved with one single call.

Use of \$count as well degrades performance. Hence, a use of \$count must also be judiciously chosen.

Therefore, for use cases where it is crucial for you to ensure that objects more than 100 have to be retrieved, you need to use paging options such as \$top and \$skip.

## Request

**URI:** /Composite/Things/PropertySet/<property set type name>/<property set ID>

**HTTP Method:** GET

**Permissions:**

<ct>.r



## Request Parameter

Parameter	Required	Data Type	Description
name	Yes	string	Name of the property set type.
id	Yes	string	Property set identifier.

## Query String Parameters

Parameter	Required	Type	Description
\$top	No	Integer	Number of records to include in the result set.
\$skip	No	Integer	Number of records to exclude from the result set.
\$count	No	Boolean	Total number of things based on the calling user's authorizations; permissible values are <i>true</i> and <i>false</i> .  If this is used in conjunction with the <i>\$top</i> and <i>\$skip</i> parameters, the result contains the total count of data available.
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; All the fields in the result set are valid.  To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.
\$select	No	String	Select condition for the data fields to be included in the result set; All the fields in the result set are valid.
\$filter	No	String	Filter condition to be applied; All the fields in the result set are valid. The filter operators supported are <i>eq</i> , <i>lt</i> , <i>gt</i> , <i>le</i> , <i>ge</i> , and <i>ne</i> .  You can use the string functions such as <i>substringof</i> , <i>startswith</i> , and <i>endswith</i> to search data using <i>\$filter</i> .

## Request Example

```
/Composite/Things/PropertySet/core.automobiles:car/car
```

## Response

**Format:** JSON

## Response Status and Error Codes

Code	Description
200	List of things with customer, location, event count, and event severity count along with the details of the specific master data property set type is retrieved.

## Payload

```
{
  "value": [{
    "_id": "<thingID>",
    "_externalId": "ABCXSeries001",
    "_name": "ABCXSeries",
    "_thingType": "core.automobiles:ABCXSeries",
    "_description": {
      "en": "Car Model ABC xSeries 001"
    },
    "_objectGroup": "473BB1AFC4A4415791543C3F2CFDD295",
    "_customer.name": "customer-test001",
    "customer.landlinePhoneNumber": "<customerLandlinePhoneNumber>",
    "customer.mobilePhoneNumber": "<customermobilePhoneNumber>",
    "customer.id": " 4D622E3DA6A14C4D87CDEC935B000002",
    "location.country.name": "country1",
    "location.country.id": "CNT1",
    "location.address": "Building1, Street Name",
    "location.id": "CD622E3DA6A14C4D87CDEC935B000001",
    "location.latitude": "53.545968",
    "location.longitude": "9.970724",
    "event.count": 100,
    "event.severity1count": 20,
    "event.severity2count": 30,
    "event.severity3count": 40,
    "propertySet": [{
      "_time": "2005-01-06T00:00:00.000Z",
      "Capacity": 1000,
      "Torque": 90,
      "Transmission": "Manual"
    }]
  }]
}
```

## 21.1.2 Read Version Information of Composite Service

This service is used to retrieve the version information of the composite service.

### Request

**URI:** /Version

**HTTP Method:** [GET](#)

**Permissions:**

<ct>.r

### Request Parameters

None

## Request Example

```
/Version
```

## Response

### Response Status and Error Codes

Code	Description
200	Version information retrieved successfully.

### Response Example

```
{
  "version" : "1.10"
}
```

## 21.1.3 HTTP Status Codes for Composites

List of status codes for composite services

In the following table, you find the list of HTTP status codes that are used by the composite services:

Code	Description
200	Success
400	Bad request (for example: header field missing, wrong filter, validation failed)
403	Forbidden (scope(s) not assigned for the user)
404	Not found (for example, data category of the property set type is not of type Master Data)
500	Internal server error

## 21.2 Composite Things: OData Service

### Overview

The composite things OData service is used to retrieve the complete information of a thing for a specific property set type and a property set. The retrieved information includes the basic details (ID, external ID, thing

type, and object group), customer details (customer name and contact), and location details (address, country, longitude, and latitude). In addition, this service retrieves information about the usage of the properties and its latest value. This service only supports the data category *MasterData*.

## OData Version: 2.0

**Root URI:** `http://<server address>[:<port number>]/CompositeThings/v1`

**Example for a Root URI in a cloud foundry environment:** `https://composite-things-odata.cfapps.eu10.hana.ondemand.com/CompositeThings`

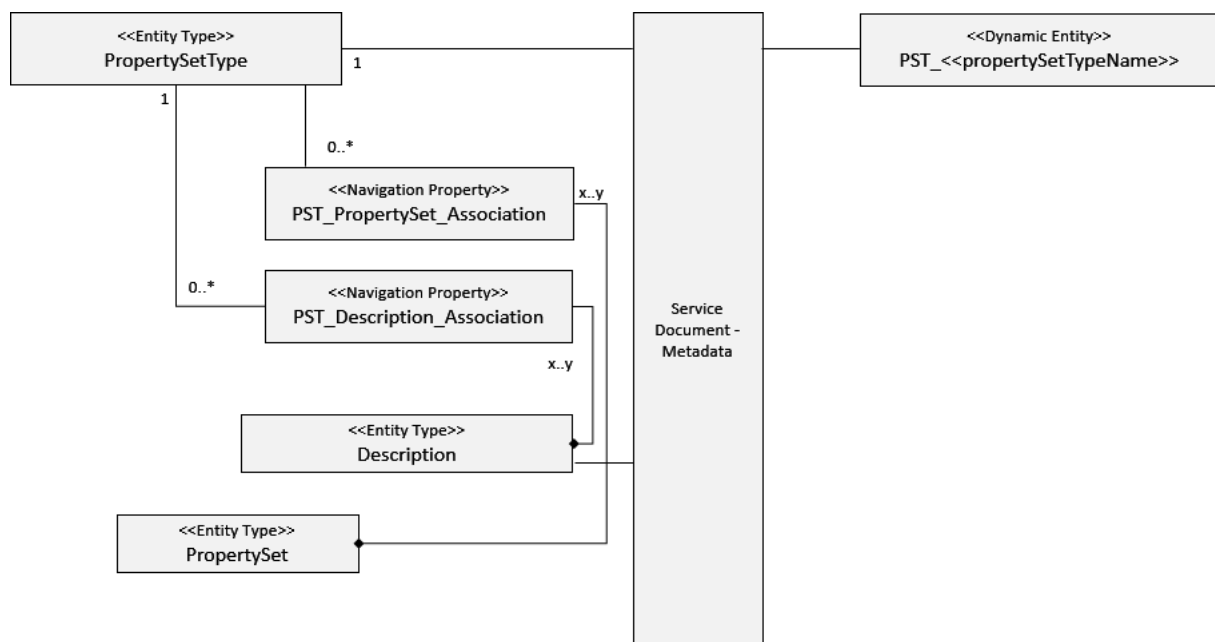
## Permissions:

`<ctodata>.SysAdmin`

`<ctodata>.r`

## Entity Data Model

The following diagram shows the EDM of Composite Things:



**Service Metadata URI:** `http://<server address>[:<port number>]/CompositeThings/v1/$metadata`

## Related Information

[Read Metadata of Composite Things \[page 1049\]](#)

[Read all Property Set Types \[page 1053\]](#)

[Read all Property Sets for a Property Set Type \[page 1056\]](#)

## 21.2.1 Read Metadata of Composite Things

Retrieves metadata of composite things

### Note

The result set does not contain the fields longitude and latitude as part of thing location details, –

- If you did not specify values for these fields while creating a **Location**.
- If you did not specify the location of a thing while creating a **Thing**.

### Request

**URI:** /CompositeThings/v1/\$metadata

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:**

<ctodata>.SysAdmin

<ctodata>.r

### Request Parameters

None

### Request Example

```
/CompositeThings/v1/$metadata
```

### Response

#### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

```
<EntityType Name="PropertySetType">
  <Key>
    <PropertyRef Name="PropertySetType"></PropertyRef>
  </Key>
  <Property Name="PropertySetType" Type="Edm.String" Nullable="false"></Property>
  <Property Name="DataCategory" Type="Edm.String"></Property>
  <Property Name="ISOCODE" Type="Edm.String"></Property>
  <Property Name="Description" Type="Edm.String"></Property>
  <NavigationProperty Name="PropertySets"
Relationship="com.sap.apptot.PST_PropertySet_Association"
FromRole="PropertySetTypeEntities" ToRole="PropertySetEntities"></NavigationProperty>
  <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.PST_Description_Association"
FromRole="PropertySetTypeEntities" ToRole="DescriptionEntities"></NavigationProperty>
</EntityType>
<EntityType Name="PropertySet">
  <Key>
    <PropertyRef Name="PropertySet"></PropertyRef>
    <PropertyRef Name="PropertySetType"></PropertyRef>
    <PropertyRef Name="ThingType"></PropertyRef>
  </Key>
  <Property Name="PropertySet" Type="Edm.String" Nullable="false"></Property>
  <Property Name="PropertySetType" Type="Edm.String" Nullable="false"></Property>
  <Property Name="ThingType" Type="Edm.String" Nullable="false"></Property>
  <Property Name="ISOCODE" Type="Edm.String"></Property>
  <Property Name="Description" Type="Edm.String"></Property>
  <Property Name="CompositeURL" Type="Edm.String"></Property>
  <NavigationProperty Name="Descriptions"
Relationship="com.sap.apptot.PS_Description_Association"
FromRole="PropertySetEntities" ToRole="DescriptionEntities"></NavigationProperty>
</EntityType>
<EntityType Name="Description">
  <Key>
    <PropertyRef Name="Language"></PropertyRef>
  </Key>
  <Property Name="Language" Type="Edm.String"></Property>
  <Property Name="Description" Type="Edm.String"></Property>
</EntityType>
<EntityType Name="PST_core_bbautomobiles__style">
  <Key>
    <PropertyRef Name="ThingId"></PropertyRef>
  </Key>
  <Property Name="ThingId" Type="Edm.String" Nullable="false"></Property>
  <Property Name="Path" Type="Edm.String"></Property>
  <Property Name="Timestamp" Type="Edm.String" Nullable="false"></Property>
  <Property Name="ThingISOCODE" Type="Edm.String"></Property>
  <Property Name="ThingDescription" Type="Edm.String"></Property>
  <Property Name="ThingName" Type="Edm.String"></Property>
  <Property Name="ThingExternalId" Type="Edm.String"></Property>
  <Property Name="ThingType" Type="Edm.String"></Property>
  <Property Name="ObjectGroup" Type="Edm.String"></Property>
  <Property Name="BPName" Type="Edm.String"></Property>
  <Property Name="BPLandlinePhoneNumber" Type="Edm.String"></Property>
  <Property Name="BPMobilePhoneNumber" Type="Edm.String"></Property>
  <Property Name="ThingCustomerId" Type="Edm.String"></Property>
  <Property Name="ThingRegionDescription" Type="Edm.String"></Property>
  <Property Name="ThingCityName" Type="Edm.String"></Property>
  <Property Name="ThingCountryDescription" Type="Edm.String"></Property>
  <Property Name="ThingCountryId" Type="Edm.String"></Property>
  <Property Name="ThingAddressRepresentation" Type="Edm.String"></Property>
```

```

        <Property Name="ThingLocationID" Type="Edm.String"></Property>
        <Property Name="Latitude" Type="Edm.Double"></Property>
        <Property Name="Longitude" Type="Edm.Double"></Property>
        <Property Name="AlternateId" Type="Edm.String"></Property>
        <Property Name="CarColor" Type="Edm.String"></Property>
    </EntityType>
    <ComplexType Name="Description">
        <Property Name="Language" Type="Edm.String"></Property>
        <Property Name="Description" Type="Edm.String"></Property>
    </ComplexType>
    <Association Name="PST_PropertySet_Association">
        <End Type="com.sap.appiot.PropertySetType" Multiplicity="1"
        Role="PropertySetTypeEntities"></End>
        <End Type="com.sap.appiot.PropertySet" Multiplicity="*"
        Role="PropertySetEntities"></End>
    </Association>
    <Association Name="PST_Description_Association">
        <End Type="com.sap.appiot.PropertySetType" Multiplicity="1"
        Role="PropertySetTypeEntities"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
        Role="DescriptionEntities"></End>
    </Association>
    <Association Name="PS_Description_Association">
        <End Type="com.sap.appiot.PropertySet" Multiplicity="1"
        Role="PropertySetEntities"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
        Role="DescriptionEntities"></End>
    </Association>
    <Association Name="PST_PropertySet_Association">
        <End Type="com.sap.appiot.PropertySetType" Multiplicity="1"
        Role="PropertySetTypeEntities"></End>
        <End Type="com.sap.appiot.PropertySet" Multiplicity="*"
        Role="PropertySetEntities"></End>
    </Association>
    <Association Name="PST_Description_Association">
        <End Type="com.sap.appiot.PropertySetType" Multiplicity="1"
        Role="PropertySetTypeEntities"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
        Role="DescriptionEntities"></End>
    </Association>
    <Association Name="PS_Description_Association">
        <End Type="com.sap.appiot.PropertySet" Multiplicity="1"
        Role="PropertySetEntities"></End>
        <End Type="com.sap.appiot.Description" Multiplicity="*"
        Role="DescriptionEntities"></End>
    </Association>
    <EntityContainer Name="CompositeThings" m:IsDefaultEntityContainer="true">
        <EntitySet Name="PropertySetTypes"
        EntityType="com.sap.appiot.PropertySetType"></EntitySet>
        <EntitySet Name="PropertySets" EntityType="com.sap.appiot.PropertySet"></
        EntitySet>
        <EntitySet Name="Descriptions" EntityType="com.sap.appiot.Description"></
        EntitySet>
        <EntitySet Name="PST_core_bbautomobiles__styles"
        EntityType="com.sap.appiot.PST_core_bbautomobiles__style"></EntitySet>
        <AssociationSet Name="PST_PropertySet_Association"
        Association="com.sap.appiot.PST_PropertySet_Association">
            <End EntitySet="PropertySetTypes" Role="PropertySetTypeEntities"></End>
            <End EntitySet="PropertySets" Role="PropertySetEntities"></End>
        </AssociationSet>
        <AssociationSet Name="PST_Description_Association"
        Association="com.sap.appiot.PST_Description_Association">
            <End EntitySet="PropertySetTypes" Role="PropertySetTypeEntities"></End>
            <End EntitySet="Descriptions" Role="DescriptionEntities"></End>
        </AssociationSet>
        <AssociationSet Name="PS_Description_Association"
        Association="com.sap.appiot.PS_Description_Association">
            <End EntitySet="PropertySets" Role="PropertySetEntities"></End>

```

```

        <End EntitySet="Descriptions" Role="DescriptionEntities"></End>
    </AssociationSet>
    <AssociationSet Name="PST_PropertySet_Association"
Association="com.sap.apptot.PST_PropertySet_Association">
        <End EntitySet="PropertySetTypes" Role="PropertySetTypeEntities"></End>
        <End EntitySet="PropertySets" Role="PropertySetEntities"></End>
    </AssociationSet>
    <AssociationSet Name="PST_Description_Association"
Association="com.sap.apptot.PST_Description_Association">
        <End EntitySet="PropertySetTypes" Role="PropertySetTypeEntities"></End>
        <End EntitySet="Descriptions" Role="DescriptionEntities"></End>
    </AssociationSet>
    <AssociationSet Name="PS_Description_Association"
Association="com.sap.apptot.PS_Description_Association">
        <End EntitySet="PropertySets" Role="PropertySetEntities"></End>
        <End EntitySet="Descriptions" Role="DescriptionEntities"></End>
    </AssociationSet>
</EntityContainer>

```

## Related Information

[Composite Things: OData Service \[page 1047\]](#)

## 21.2.2 Read Details of a Property Set Type

Retrieves details of a specific property set type

### Request

**URI:** /CompositeThings/v1/PropertySetTypes('<property set type name>')

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:**

<ctodata>.SysAdmin

<ctodata>.r

### Request Parameters

Parameter	Required	Data Type	Description
Property set type name	Yes	String	Name of the property set type

### Request Example

```
/CompositeThings/v1/PropertySetTypes('core.automobiles:style')
```



## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [XML](#)

Media types supported are [JSON](#) and [XML](#). The [XML](#) for this method has the following structure:

```
<link href="PropertySetTypes('core.autobomobiles%3Astyle')" rel="edit"
title="PropertySetType"></link>
<link href="PropertySetTypes('core.autobomobiles%3Astyle')/PropertySets"
rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/PropertySets"
title="PropertySets" type="application/atom+xml;type=feed"></link>
<link href="PropertySetTypes('core.autobomobiles%3Astyle')/Descriptions"
rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/Descriptions"
title="Descriptions" type="application/atom+xml;type=feed"></link>
<content type="application/xml">
  <m:properties>
    <d:PropertySetType>core.autobomobiles:style</d:PropertySetType>
    <d:DataCategory>MasterData</d:DataCategory>
    <d:ISOCODE>en_US</d:ISOCODE>
    <d:Description m:null="true"></d:Description>
  </m:properties>
</content>
</entry>
```

## Related Information

[Composite Things: OData Service \[page 1047\]](#)

### 21.2.3 Read all Property Set Types

Retrieves all property set types

By default, the service displays a maximum of 100 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** /CompositeThings/v1/PropertySetTypes

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:**

<ctodata>.SysAdmin

<ctodata>.r

## Request Parameters

None

## Request Example

```
/CompositeThings/v1/PropertySetTypes
```

## Query String Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; only valid field is the <a href="#">PropertySetType</a> . To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$filter	No	String	Filter condition to be applied; the only valid field is <a href="#">PropertySetType</a> . The filter operators supported are <b>eq</b> , <b>lt</b> , <b>gt</b> , <b>le</b> , <b>ge</b> , and <b>ne</b> . Multiple filters are supported using the <a href="#">AND</a> separator.

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [XML](#)

Media types supported are [JSON](#) and [XML](#). The [XML](#) for this method has the following structure:

```
<link href="PropertySetTypes('core.autobomobiles%3Astyle')" rel="edit"
title="PropertySetType"></link>
<link href="PropertySetTypes('core.autobomobiles%3Astyle')/PropertySets"
rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/PropertySets"
title="PropertySets" type="application/atom+xml;type=feed"></link>
<link href="PropertySetTypes('core.autobomobiles%3Astyle')/Descriptions"
rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/Descriptions"
title="Descriptions" type="application/atom+xml;type=feed"></link>
<content type="application/xml">
  <m:properties>
    <d:PropertySetType>core.autobomobiles:style</d:PropertySetType>
    <d:DataCategory>MasterData</d:DataCategory>
    <d:ISOCODE>en_US</d:ISOCODE>
    <d:Description m:null="true"></d:Description>
  </m:properties>
</content>
</entry>
<entry>
  <link href="PropertySetTypes('core.autobomobiles%3ACarInfo')" rel="edit"
title="PropertySetType"></link>
  <link href="PropertySetTypes('core.autobomobiles%3ACarInfo')/
PropertySets" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/
PropertySets" title="PropertySets" type="application/atom+xml;type=feed"></link>
  <link href="PropertySetTypes('core.autobomobiles%3ACarInfo')/
Descriptions" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/
Descriptions" title="Descriptions" type="application/atom+xml;type=feed"></link>
  <content type="application/xml">
    <m:properties>
      <d:PropertySetType>core.autobomobiles:CarInfo</d:PropertySetType>
      <d:DataCategory>MasterData</d:DataCategory>
      <d:ISOCODE>en_US</d:ISOCODE>
      <d:Description m:null="true"></d:Description>
    </m:properties>
  </content>
</entry>
```

## Related Information

[Composite Things: OData Service \[page 1047\]](#)

## 21.2.4 Read all Property Sets for a Property Set Type

Retrieves all property sets for the specified property set type

### Request

**URI:** /CompositeThings/v1/PropertySetTypes('<property set type name>')/PropertySets

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:**

<ctodata>.SysAdmin

<ctodata>.r

### Request Parameters

Parameter	Required	Data Type	Description
property set type name	Yes	String	Name of the property set type

### Request Example

```
/CompositeThings/v1/PropertySetTypes('core.automobiles:style')/PropertySets
```

### Query String Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; only valid field is the <a href="#">property set type name</a> . To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.

Parameter	Required	Data Type	Description
\$filter	No	String	Filter condition to be applied; the only valid field is <i>name</i> . The filter operators supported are <i>eq</i> , <i>lt</i> , <i>gt</i> , <i>le</i> , <i>ge</i> , and <i>ne</i> . Multiple filters are supported using the <i>AND</i> separator. Only valid field is <i>property set type name</i> .

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [XML](#)

Media types supported are [JSON](#) and [XML](#). The [XML](#) for this method has the following structure:

```
<link href="PropertySets(PropertySet='carstyle',PropertySetType='core.automobiles%3Astyle',ThingType='core.automobiles%3ABCXSeries') " rel="edit" title="PropertySet"></link>
<link href="PropertySets(PropertySet='carstyle',PropertySetType='core.automobiles%3Astyle',ThingType='core.automobiles%3ABCXSeries')/Descriptions" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/Descriptions" title="Descriptions" type="application/atom+xml;type=feed"></link>
<content type="application/xml">
  <m:properties>
    <d:PropertySet>carstyle</d:PropertySet>
    <d:PropertySetType>core.automobiles:style</d:PropertySetType>
    <d:ThingType>core.automobiles:ABCXSeries</d:ThingType>
    <d:ISOCODE>en</d:ISOCODE>
    <d:Description>Description of the property set type</d:Description>
    <d:CompositeURL>PST_core_automobiles__style?$filter=Path eq 'carstyle'</d:CompositeURL>
  </m:properties>
</content>
```

## Related Information

[Composite Things: OData Service \[page 1047\]](#)

## 21.2.5 Read Things for Property Set of a Property Set Type

Retrieves a list of unique things that uses property set of the specified property set type

### Note

The result set does not contain the fields longitude and latitude as part of thing location details, –

- If you did not specify values for these fields while creating a **Location**.
- If you did not specify the location of a thing while creating a **Thing**.

### Request

**URI:** /CompositeThings/v1/<property set type name>?\$filter=Path eq '<property set ID>'

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:**

<ctodata>.SysAdmin

<ctodata>.r

### Request Parameters

Parameter	Required	Data Type	Description
Property set type name	Yes	String	Entity set name of the property set type (you can retrieve this from the metadata)
Property set ID	Yes	String	Unique identifier of the property type (that belongs to the property set type) defined in a thing type

### Request Example

```
/CompositeThings/v1/PST_core_automobiles__style?$filter=Path eq 'carstyle'
```

### Query String Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set
\$skip	No	Integer	Number of the first n records to be excluded from the result set.

Parameter	Required	Data Type	Description
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; Valid fields are <i>thing ID</i>, <i>thing name</i>, <i>external ID</i>, and <i>timestamp</i>.</p> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; Valid fields are <i>thing ID</i>, <i>thing name</i>, <i>external ID</i>, and <i>timestamp</i>.</p> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; the only valid field is name.</p> <p>The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the <i>AND</i> separator. Valid fields are <i>thingID</i>, <i>thing name</i>, <i>external ID</i>, and <i>timestamp</i>.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; all fields in the result set are valid.</p>

### Note

The date functions such as *day*, *hour*, *minute*, *second*, *month*, and *year* are supported in *\$filter* query. For more information, see [Filter Conditions \(\\$filter\)](#) [page 1214].

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: *XML*

Media types supported are *JSON* and *XML*. The *XML* for this method has the following structure:

```
<link href="PST_core_automobiles__styles" rel="self"
title="PST_core_automobiles__styles" />
  <entry>
    <id>https://composite-things-odata-ci.cfapps.staging.sic.ondemand.com:443/
CompositeThings/v1/
PST_core_automobiles__carstyles('E601CF9BC9B6479281E3628B08DABE42')</id>
    <title type="text">PST_core_automobiles__styles</title>
    <updated>2016-11-19T09:38:16.614Z</updated>
    <category term="com.sap.appiot.PST_core_automobiles__style" scheme="http://
schemas.microsoft.com/ado/2007/08/dataservices/scheme" />
    <link
href="PST_core_automobiles__styles('E601CF9BC9B6479281E3628B08DABE42')"
rel="edit" title="PST_core_automobiles__style" />
    <link
href="PST_core_automobiles__styles('E601CF9BC9B6479281E3628B08DABE42')/
Descriptions" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/
Descriptions" title="Descriptions" type="application/atom+xml;type=feed" />
    <content type="application/xml">
      <m:properties>
        <d:ThingId>E601CF9BC9B6479281E3628B08DABE42</d:ThingId>
        <d:Path>/carstyle</d:Path>
        <d:Timestamp>1970-01-01T00:00:00.000Z</d:Timestamp>
        <d:ThingISOCODE>en</d:ThingISOCODE>
        <d:ThingDescription>car thing 1 description</d:ThingDescription>
        <d:ThingName>carthing1</d:ThingName>
        <d:ThingExternalId>carthingextid </d:ThingExternalId>
        <d:ThingType>core.automobiles:ABCXSeries</d:ThingType>
        <d:ObjectGroup>3F2EE14BEF184BE7A1BDEBC12CC7409B</d:ObjectGroup>
        <d:BPName m:null="true" />
        <d:BPLandlinePhoneNumber m:null="true" />
        <d:BPMobilePhoneNumber m:null="true" />
        <d:ThingCustomerId m:null="true" />
        <d:ThingRegionDescription m:null="true" />
        <d:ThingCityName m:null="true" />
        <d:ThingCountryDescription m:null="true" />
        <d:ThingCountryId m:null="true" />
        <d:ThingAddressRepresentation m:null="true" />
        <d:ThingLocationID m:null="true" />
        <d:AlternateId>carthinglaltid</d:AlternateId>
        <d:Latitude>0.0</d:Latitude>
        <d:Longitude>0.0</d:Longitude>
        <d:color>blue</d:color>
        <d:model>HZ1021</d:model>
      </m:properties>
    </content>
  </entry>
</entry>
```



```

    <id>https://composite-things-odata-ci.cfapps.staging.sic.ondemand.com:443/
CompositeThings/v1/
PST_core_automobiles__styles('B165AC277A85444DB29BF7FED08AAE08')</id>
    <title type="text">PST_core_automobiles__styles</title>
    <updated>2016-11-19T09:38:16.614Z</updated>
    <category term="com.sap.appiot.PST_core_automobiles__style" scheme="http://
schemas.microsoft.com/ado/2007/08/dataservices/scheme" />
    <link
href="PST_core_automobiles__styles('B165AC277A85444DB29BF7FED08AAE08') "
rel="edit" title="PST_core_automobiles__style" />
    <link
href="PST_core_automobiles__styles('B165AC277A85444DB29BF7FED08AAE08')/
Descriptions" rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/
Descriptions" title="Descriptions" type="application/atom+xml;type=feed" />
    <content type="application/xml">
      <m:properties>
        <d:ThingId>B165AC277A85444DB29BF7FED08AAE08</d:ThingId>
        <d:Path m:null="true" />
        <d:Timestamp>1970-01-01T00:00:00.000Z</d:Timestamp>
        <d:ThingISOCODE>en</d:ThingISOCODE>
        <d:ThingDescription>car thing 2 description</d:ThingDescription>
        <d:ThingName>carthing2</d:ThingName>
        <d:ThingExternalId>carthingextid </d:ThingExternalId>
        <d:ThingType>core.automobiles:ABCXSeries</d:ThingType>
        <d:ObjectGroup>3F2EE14BEF184BE7A1BDEBC12CC7409B</d:ObjectGroup>
        <d:BPName m:null="true" />
        <d:BPlandlinePhoneNumber m:null="true" />
        <d:BPMobilePhoneNumber m:null="true" />
        <d:ThingCustomerId m:null="true" />
        <d:ThingRegionDescription m:null="true" />
        <d:ThingCityName m:null="true" />
        <d:ThingCountryDescription m:null="true" />
        <d:ThingCountryId m:null="true" />
        <d:ThingAddressRepresentation m:null="true" />
        <d:ThingLocationID m:null="true" />
        <d:AlternateId>carthing2altid</d:AlternateId>
        <d:Latitude>0.0</d:Latitude>
        <d:Longitude>0.0</d:Longitude>
        <d:color m:null="true" />
        <d:model m:null="true" />
      </m:properties>
    </content>
  </entry>

```

## Related Information

[Composite Things: OData Service \[page 1047\]](#)

## 21.3 Composite Events: OData Service

### Overview

The composite events OData service is used to retrieve all user-defined event details for the specified event type. In addition, the service also retrieves the event data.

#### OData Version: 2.0

**Root URI:** `http://<server address>[:<port number>]/CompositeEvents/v1`

**Example for a Root URI in a cloud foundry environment:** `https://composite-events-odata.cfapps.eu10.hana.ondemand.com/CompositeEvents`

#### Permissions:

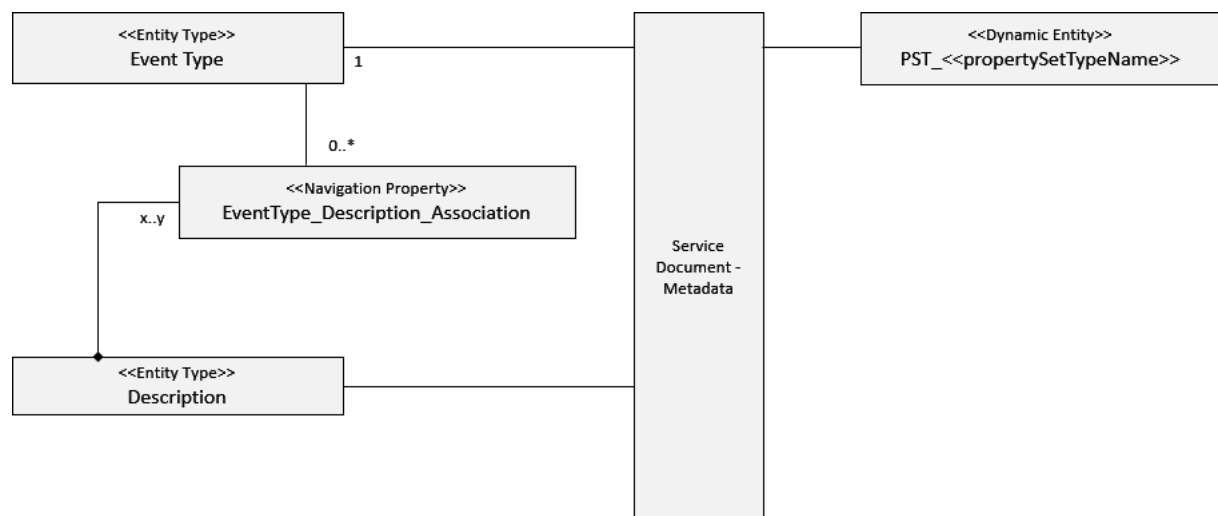
`<ctodata>.SysAdmin`

`<ctodata>.r`

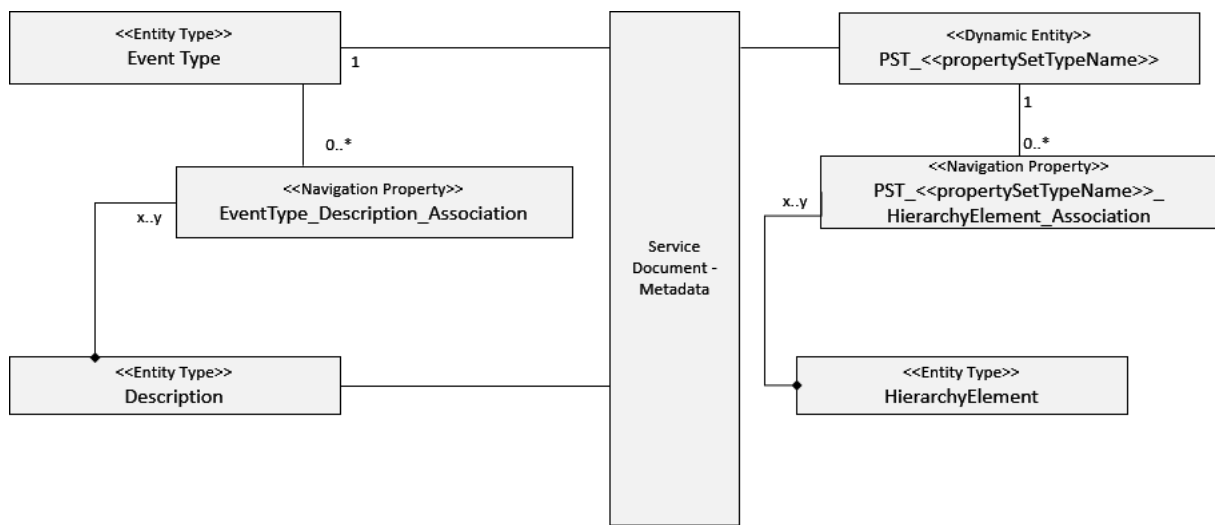
### Entity Data Model

The following diagrams illustrate the EDM of composite events OData services for different versions:

#### Version 1



#### Version 2



**Service Metadata URI:** `http://<server address>[:<port number>]/CompositeEvents/v1/$metadata`

## Related Information

[Read Metadata of Composite Events \[page 1063\]](#)

[Read Details of all Event Types \[page 1067\]](#)

[Read Details of an Event Type \[page 1071\]](#)

[Read Events and Event Data for Property Set of a Property Set Type \[page 1073\]](#)

[Read Details of all Events for an Event Type \[page 1080\]](#)

[Read Details of an Event for an Event Type \[page 1089\]](#)

## 21.3.1 Read Metadata of Composite Events

With this method, you can retrieve metadata of composite events.

### Request

**URI:**

Version	URI
v1	<code>/CompositeEvents/v1/\$metadata</code>
v2	<code>/CompositeEvents/v2/\$metadata</code>

**Operation Type:** CRUD

HTTP Method: [GET](#)

#### Permissions:

<ctodata>.SysAdmin

<ctodata>.r

#### Request Parameters

None

#### Request Example

```
/CompositeEvents/v1/$metadata
```

## Response

#### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [XML](#) for this method has the structure as depicted in the following examples:

The request GET /CompositeEvents/v1/\$metadata retrieves the following example:

```
<EntityType Name="EventType">
  <Key>
    <PropertyRef Name="EventType"/>
  </Key>
  <Property Name="EventType" Type="Edm.String" Nullable="false"/>
  <Property Name="EventTypeDescription" Type="Edm.String"/>
  <Property Name="PropertySet" Type="Edm.String"/>
  <Property Name="PropertySetType" Type="Edm.String"/>
  <Property Name="CompositeURL" Type="Edm.String"/>
  <NavigationProperty Name="Descriptions"
Relationship="com.abc.apptot.EventType_Description_Association"
FromRole="EventTypeEntities" ToRole="DescriptionEntities"/>
</EntityType>
<EntityType Name="PST_com_abc_core__ExceededTemperature">
  <Key>
    <PropertyRef Name="EventId"/>
  </Key>
  <Property Name="EventId" Type="Edm.String" Nullable="false"/>
  <Property Name="Path" Type="Edm.String"/>
  <Property Name="CorrelationId" Type="Edm.String"/>
  <Property Name="BusinessTimestamp" Type="Edm.String"/>
  <Property Name="EventType" Type="Edm.String"/>
</EntityType>
```

```

    <Property Name="Status" Type="Edm.String"/>
    <Property Name="Severity" Type="Edm.String"/>
    <Property Name="Code" Type="Edm.String"/>
    <Property Name="ThingId" Type="Edm.String"/>
    <Property Name="ThingProperty" Type="Edm.String"/>
    <Property Name="ExternalId" Type="Edm.String"/>
    <Property Name="PropertySet" Type="Edm.String"/>
    <Property Name="Temperature" Type="Edm.Int64"/>
</EntityType>
<EntityType Name="PST_com_abc_core__ExceededPressure">
    <Key>
        <PropertyRef Name="EventId"/>
    </Key>
    <Property Name="EventId" Type="Edm.String" Nullable="false"/>
    <Property Name="Path" Type="Edm.String"/>
    <Property Name="CorrelationId" Type="Edm.String"/>
    <Property Name="BusinessTimestamp" Type="Edm.String"/>
    <Property Name="EventType" Type="Edm.String"/>
    <Property Name="Status" Type="Edm.String"/>
    <Property Name="Severity" Type="Edm.String"/>
    <Property Name="Code" Type="Edm.String"/>
    <Property Name="ThingId" Type="Edm.String"/>
    <Property Name="ThingProperty" Type="Edm.String"/>
    <Property Name="ExternalId" Type="Edm.String"/>
    <Property Name="PropertySet" Type="Edm.String"/>
    <Property Name="Pressure" Type="Edm.Int64"/>
</EntityType>
<EntityType Name="Description">
    <Key>
        <PropertyRef Name="FieldName"></PropertyRef>
    </Key>
    <Property Name="FieldName" Type="Edm.String"></Property>
    <Property Name="Value" Type="Edm.String"></Property>
</EntityType>

```

The request GET /CompositeEvents/v2/\$metadata retrieves the following example:

```

<EntityType Name="EventType">
    <Key>
        <PropertyRef Name="EventType"/>
    </Key>
    <Property Name="EventType" Type="Edm.String" Nullable="false"/>
    <Property Name="EventTypeDescription" Type="Edm.String"/>
    <Property Name="PropertySet" Type="Edm.String"/>
    <Property Name="PropertySetType" Type="Edm.String"/>
    <Property Name="CompositeURL" Type="Edm.String"/>
    <NavigationProperty Name="Descriptions"
Relationship="com.sap.apiot.EventType_Description_Association"
FromRole="EventTypeEntities" ToRole="DescriptionEntities"/>
</EntityType>

<EntityType Name="PST_com_sap_core__ExceededTemperature">
    <Key>
        <PropertyRef Name="EventId"></PropertyRef>
    </Key>
    <Property Name="EventId" Type="Edm.String" Nullable="false"></Property>
    <Property Name="ExternalId" Type="Edm.String"></Property>
    <Property Name="Path" Type="Edm.String"></Property>
    <Property Name="BusinessTimestamp" Type="Edm.DateTimeOffset"
Nullable="false"></Property>
    <Property Name="CorrelationId" Type="Edm.String"></Property>
    <Property Name="EventType" Type="Edm.String"></Property>
    <Property Name="Status" Type="Edm.String"></Property>
    <Property Name="Severity" Type="Edm.Int16"></Property>
    <Property Name="Code" Type="Edm.String"></Property>
    <Property Name="ThingId" Type="Edm.String"></Property>
    <Property Name="ThingExternalId" Type="Edm.String"></Property>

```

```

    <Property Name="ThingProperty" Type="Edm.String"></Property>
    <Property Name="ThingISOCODE" Type="Edm.String"></Property>
    <Property Name="ThingDescription" Type="Edm.String"></Property>
    <Property Name="ThingName" Type="Edm.String"></Property>
    <Property Name="ThingType" Type="Edm.String"></Property>
    <Property Name="BPName" Type="Edm.String"></Property>
    <Property Name="BPLandlinePhoneNumber" Type="Edm.String"></Property>
    <Property Name="BPMobilePhoneNumber" Type="Edm.String"></Property>
    <Property Name="ThingCustomerId" Type="Edm.String"></Property>
    <Property Name="ThingRegionDescription" Type="Edm.String"></Property>
    <Property Name="ThingCityName" Type="Edm.String"></Property>
    <Property Name="ThingCountryDescription" Type="Edm.String"></Property>
    <Property Name="ThingCountryId" Type="Edm.String"></Property>
    <Property Name="ThingAddressRepresentation" Type="Edm.String"></
Property>
    <Property Name="ThingLocationId" Type="Edm.String"></Property>
    <Property Name="Temperature" Type="Edm.Int64"/>
</EntityType>

<EntityType Name="PST_com_sap_core__ExceededPressure">
    <Key>
        <PropertyRef Name="EventId"></PropertyRef>
    </Key>
    <Property Name="EventId" Type="Edm.String" Nullable="false"></Property>
    <Property Name="ExternalId" Type="Edm.String"></Property>
    <Property Name="Path" Type="Edm.String"></Property>
    <Property Name="BusinessTimestamp" Type="Edm.DateTimeOffset"
Nullable="false"></Property>
    <Property Name="CorrelationId" Type="Edm.String"></Property>
    <Property Name="EventType" Type="Edm.String"></Property>
    <Property Name="Status" Type="Edm.String"></Property>
    <Property Name="Severity" Type="Edm.Int16"></Property>
    <Property Name="Code" Type="Edm.String"></Property>
    <Property Name="ThingId" Type="Edm.String"></Property>
    <Property Name="ThingExternalId" Type="Edm.String"></Property>
    <Property Name="ThingProperty" Type="Edm.String"></Property>
    <Property Name="ThingISOCODE" Type="Edm.String"></Property>
    <Property Name="ThingDescription" Type="Edm.String"></Property>
    <Property Name="ThingName" Type="Edm.String"></Property>
    <Property Name="ThingType" Type="Edm.String"></Property>
    <Property Name="BPName" Type="Edm.String"></Property>
    <Property Name="BPLandlinePhoneNumber" Type="Edm.String"></Property>
    <Property Name="BPMobilePhoneNumber" Type="Edm.String"></Property>
    <Property Name="ThingCustomerId" Type="Edm.String"></Property>
    <Property Name="ThingRegionDescription" Type="Edm.String"></Property>
    <Property Name="ThingCityName" Type="Edm.String"></Property>
    <Property Name="ThingCountryDescription" Type="Edm.String"></Property>
    <Property Name="ThingCountryId" Type="Edm.String"></Property>
    <Property Name="ThingAddressRepresentation" Type="Edm.String"></
Property>
    <Property Name="ThingLocationId" Type="Edm.String"></Property>
    <Property Name="Pressure" Type="Edm.Int64"/>
</EntityType>

<EntityType Name="Description">
    <Key>
        <PropertyRef Name="FieldName"></PropertyRef>
    </Key>
    <Property Name="FieldName" Type="Edm.String"></Property>
    <Property Name="Value" Type="Edm.String"></Property>
</EntityType>

```

## Related Information

[Composite Events: OData Service \[page 1062\]](#)

## 21.3.2 Read Details of all Event Types

### Request

URI:

Version	URI
v1	/CompositeEvents/v1/EventTypes
v2	/CompositeEvents/v2/EventTypes

By default, the service displays a maximum of 100 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:**

<ctodata>.SysAdmin

<ctodata>.r

### Request Parameters

None

### Request Example

```
/CompositeEvents/v1/EventTypes
```

### Query String Parameters

Parameter	Required	Data Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set

Parameter	Required	Data Type	Description
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; valid fields are <i>event type name</i>, <i>property set type name</i>, and <i>property set ID</i>.</p> <p>To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; the only valid field is <i>name</i>. The filter operators supported are <i>eq</i>, <i>lt</i>, <i>gt</i>, <i>le</i>, <i>ge</i>, and <i>ne</i>.</p> <p>Multiple filters are supported using the <i>AND</i> separator.</p> <p>Valid fields are <i>event type name</i>, <i>property set type name</i>, and <i>property set ID</i>.</p>
\$select	No	String	<p>Select condition for the data fields to be included in the result set; valid fields are <i>event type name</i>, <i>property set type name</i>, <i>property set ID</i>, and <i>composite URL</i>. Event type name is a mandatory field. However, if you specify composite URL in the select condition, you must specify both <i>property set type name</i> and <i>property set ID</i> along with the <i>event type name</i>.</p>

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully



## Payload

Format: *XML*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as depicted in the following examples:

### Example 1

The request `/CompositeEvents/v1/EventTypes` retrieves all event types to which the user has access to.

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedSpeed')",
          "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedSpeed')",
          "type": "com.sap.apptot.EventType"
        },
        "EventType": "core.automobiles:exceedSpeed",
        "EventTypeDescription": "Event type for exceeded speed",
        "PropertySet": "wheelspeed",
        "PropertySetType": "core.automobiles:wheelEPST",
        "CompositeURL": "PST_core_automobiles__wheelEPSTs?$filter=Path eq 'wheelspeed'",
        "Descriptions": {
          "__deferred": {
            "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedSpeed')/Descriptions"
          }
        }
      },
      {
        "__metadata": {
          "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedTemp')",
          "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedTemp')",
          "type": "com.sap.apptot.EventType"
        },
        "EventType": "core.automobiles:exceedTemp",
        "EventTypeDescription": "Event type for exceeded temperature",
        "PropertySet": "wheeltemp",
        "PropertySetType": "core.automobiles:wheelEPST",
        "CompositeURL": "PST_core_automobiles__wheelEPSTs?$filter=Path eq 'wheeltemp'",
        "Descriptions": {
          "__deferred": {
            "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedTemp')/Descriptions"
          }
        }
      }
    ]
  }
}
```

## Example 2

The request `/CompositeEvents/v1/EventTypes?$expand=Descriptions` retrieves all event types to which the user has access to along with the expanded view of descriptions.

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedSpeed')",
          "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedSpeed')",
          "type": "com.sap.apptot.EventType"
        },
        "EventType": "core.automobiles:exceedSpeed",
        "EventTypeDescription": "Event type for exceeded speed",
        "PropertySet": "wheelspeed",
        "PropertySetType": "core.automobiles:wheelEPST",
        "CompositeURL": "PST_core_automobiles__wheelEPSTs?$filter=Path eq 'wheelspeed'",
        "Descriptions": {
          "results": [
            {
              "__metadata": {
                "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/Descriptions('en')",
                "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/Descriptions('en')",
                "type": "com.sap.apptot.Description"
              },
              "Language": "en",
              "Description": "Event type for exceeded speed"
            }
          ]
        }
      },
      {
        "__metadata": {
          "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedTemp')",
          "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedTemp')",
          "type": "com.sap.apptot.EventType"
        },
        "EventType": "core.automobiles:exceedTemp",
        "EventTypeDescription": "Event type for exceeded temperature",
        "PropertySet": "wheeltemp",
        "PropertySetType": "core.automobiles:wheelEPST",
        "CompositeURL": "PST_core_automobiles__wheelEPSTs?$filter=Path eq 'wheeltemp'",
        "Descriptions": {
          "results": [
            {
              "__metadata": {
                "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/Descriptions('en')",
                "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/Descriptions('en')",
                "type": "com.sap.apptot.Description"
              },
              "Language": "en",
              "Description": "Event type for exceeded temperature"
            }
          ]
        }
      }
    ]
  }
}
```

```

    "Description": "Event type for exceeded temperature"
  }
}
]
}
]
}
}

```

## Related Information

[Composite Events: OData Service \[page 1062\]](#)

### 21.3.3 Read Details of an Event Type

#### Request

URI:

Version	URI
v1	/CompositeEvents/v1/EventTypes('<event type name>')
v2	/CompositeEvents/v2/EventTypes('<event type name>')

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:**

<ctodata>.SysAdmin

<ctodata>.r

#### Request Parameters

Parameter	Required	Data Type	Description
Event type name	Yes	String	Name of the event type

#### Request Example

```
/CompositeEvents/v1/EventTypes('core.automobiles:exceedTemp')
```

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: *XML*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as shown in the following examples:

### Example 1

The request `/CompositeEvents/v1/EventTypes('core.automobiles:exceedTemp')` retrieves the details of a specific event type.

```
{
  "d": {
    "__metadata": {
      "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedTemp')",
      "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedTemp')",
      "type": "com.sap.apptot.EventType"
    },
    "EventType": "core.automobiles:exceedTemp",
    "EventTypeDescription": "Event type for exceeded temperature",
    "PropertySet": "wheeltemp",
    "PropertySetType": "core.automobiles:wheelEPST",
    "CompositeURL": "PST_core_automobiles__wheelEPSTs?$filter=Path eq 'wheeltemp'",
    "Descriptions": {
      "__deferred": {
        "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedTemp')/Descriptions"
      }
    }
  }
}
```

### Example 2

The request `/CompositeEvents/v1/EventTypes('core.automobiles:exceedTemp')?$expand=Descriptions` retrieves the details of a specific event type.

```
{
  "d": {
    "__metadata": {
      "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/EventTypes('core.automobiles%3AexceedTemp')",

```

```

        "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/
EventTypes('core.automobiles%3AexceedTemp')",
        "type": "com.sap.apptot.EventType"
    },
    "EventType": "core.automobiles:exceedTemp",
    "EventTypeDescription": "Event type for exceeded temperature",
    "PropertySet": "wheeltemp",
    "PropertySetType": "core.automobiles:wheelEPST",
    "CompositeURL": "PST_core_automobiles__wheelEPST1s?$filter=Path eq
'wheeltemp'",
    "Descriptions": {
        "results": [{
            "__metadata": {
                "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/Descriptions('en')",
                "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/Descriptions('en')",
                "type": "com.sap.apptot.Description"
            },
            "Language": "en",
            "Description": "Event type for exceeded temperature"
        }]
    }
}

```

## Related Information

[Composite Events: OData Service \[page 1062\]](#)

### 21.3.4 Read Events and Event Data for Property Set of a Property Set Type

Retrieves all events and event data for the specified property set type and property set

#### **i** Note

With this service, you can retrieve events (with same correlation ID or different correlation ID) with latest time stamp for every correlation ID.

By default, the service displays a maximum of 100 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

URI:

Version	URI
v1	/CompositeEvents/v1/CompositeURL
v2	/CompositeEvents/v2/CompositeURL

### Note

You will find the value for the [CompositeURL](#) field in the response payload of /CompositeEvents/v1/EventTypes('core.automobiles:exceedTemp')

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:**

<ctodata>.SysAdmin

<ctodata>.r

### Request Example

```
/CompositeEvents/v1/PST_core_automobiles__wheelEPSTs?$filter=Path eq 'wheeltemp'
```

### Query String Parameters

Parameter	Required	Data Type	Description
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; all fields in the result set are valid. In addition, all dynamic columns are valid.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$filter	No	String	Filter condition to be applied; all fields in the result set are valid. However, for /v2 version, the field <a href="#">BusinessTimestamp</a> is not supported.  The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> .  Multiple filters are supported using the <a href="#">and</a> separator.

Parameter	Required	Data Type	Description
\$select	No	String	Select condition for the data fields to be included in the result set; all fields in the result set are valid. In addition, all dynamic columns are valid.

### Note

The date functions such as [day](#), [hour](#), [minute](#), [second](#), [month](#), and [year](#) are supported in `$filter` query. For more information, see [Filter Conditions \(\\$filter\)](#) [page 1214].

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [XML](#)

Media types [JSON](#) and [XML](#) are supported. The [JSON](#) for this method has the structure as depicted in the following examples.

### Example 1

The request `/CompositeEvents/v1/PST_core_automobiles__wheelEPSTs?$filter=Path eq 'wheeltemp'`, retrieves the following result:

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/PST_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
          "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/PST_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
          "type": "com.sap.apptot.PST_core_automobiles__wheelEPST"
        },
        "EventId": "2C0108673D43483B94AFEB337646CF9",
        "Path": "wheeltemp",
        "BusinessTimestamp": "2018-08-05 09:00:00.000000000",
        "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
        "EventType": "core.automobiles:exceedTemp",
        "Type": "Danger",
        "Status": "Open",
      }
    ]
  }
}
```

```

        "Severity": "1",
        "Code": null,
        "ThingId": "71A184C1345C4DD68D46315181409738",
        "ThingProperty": "core.automobiles:ABCXSeries/frwheel",
        "ExternalId": "exceedtemp-extID",
        "Description": "Alert on exceeded temperature",
        "Source": null,
        "Data": null,
        "application": "exceeded temperature app",
        "temperature": "55",
        "message": "Alert on exceeded temperature"
    },
    {
        "__metadata": {
            "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/
PST_core_automobiles__wheelEPSTs('2C9A8540B2A24560AA9CFB2EC209469D')",
            "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v1/
PST_core_automobiles__wheelEPSTs('2C9A8540B2A24560AA9CFB2EC209469D')",
            "type": "PST_core_automobiles__wheelEPST"
        },
        "EventId": "2C9A8540B2A24560AA9CFB2EC209469D",
        "Path": "wheeltemp",
        "BusinessTimestamp": "2018-06-05 09:00:00.000000000",
        "CorrelationId": "a836f37ef69c4b75a49769bf17ec560f",
        "EventType": "core.automobiles:exceedTemp",
        "Type": "Warning",
        "Status": "Open",
        "Severity": "2",
        "Code": null,
        "ThingId": "71A184C1345C4DD68D46315181409738",
        "ThingProperty": "core.automobiles:ABCXSeries/frwheel",
        "ExternalId": "exceedtemp-extID",
        "Description": "Alert on exceeded temperature",
        "Source": null,
        "Data": null,
        "application": "exceeded temperature app",
        "temperature": "35",
        "message": "Alert on exceeded temperature"
    }
]
}

```

## Example 2

The request `/CompositeEvents/v2/PST_core_automobiles__wheelEPSTs?$filter=Path eq 'wheeltemp'`, retrieves the following result:

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/
PST_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
          "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/
PST_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
          "type": "com.sap.appiot.PST_core_automobiles__wheelEPST"
        },
        "EventId": "2C0108673D43483B94AFEB337646CF9",
        "ExternalId": "exceedtemp-extID",
        "Path": "wheeltemp",
        "BusinessTimestamp": "/Date(1533459600000)/",

```



```

        "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
        "EventType": "core.automobiles:exceedTemp",
        "Type": "Danger",
        "Status": "Open",
        "StatusDescription": "Open Status",
        "Severity": 1,
        "SeverityDescription": "High",
        "Code": null,
        "CodeDescription": null,
        "ThingId": "71A184C1345C4DD68D46315181409738",
        "ThingExternalId": "ABCXSeries-extID",
        "ThingProperty": "core.automobiles:ABCXSeries/frwheel",
        "ThingISOCODE": "en",
        "ThingDescription": "ABCXSeries thing description",
        "ThingName": "ABCXSeries",
        "ThingType": "core.automobiles:ABCXSeries",
        "BPName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCityName": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingAddressRepresentation": null,
        "ThingLocationId": null,
        "EventInfo": "Alert on exceeded temperature",
        "application": "exceeded temperature app",
        "temperature": "55",
        "message": "Alert on exceeded temperature"
    },
    {
        "__metadata": {
            "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/
PST_core_automobiles__wheelEPSTs('2C9A8540B2A24560AA9CFB2EC209469D')",
            "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/
PST_core_automobiles__wheelEPSTs('2C9A8540B2A24560AA9CFB2EC209469D')",
            "type": "com.sap.apptot.PST_core_automobiles__wheelEPST"
        },
        "EventId": "2C9A8540B2A24560AA9CFB2EC209469D",
        "ExternalId": "exceedtemp-extID",
        "Path": "wheeltemp",
        "BusinessTimestamp": "/Date(1528189200000)/",
        "CorrelationId": "a836f37ef69c4b75a49769bf17ec560f",
        "EventType": "core.automobiles:exceedTemp",
        "Type": "Warning",
        "Status": "Open",
        "StatusDescription": "Open Status",
        "Severity": 2,
        "SeverityDescription": "Medium",
        "Code": null,
        "CodeDescription": null,
        "ThingId": "71A184C1345C4DD68D46315181409738",
        "ThingExternalId": "ABCXSeries-extID",
        "ThingProperty": "core.automobiles:ABCXSeries/frwheel",
        "ThingISOCODE": "en",
        "ThingDescription": "ABCXSeries thing description",
        "ThingName": "ABCXSeries",
        "ThingType": "core.automobiles:ABCXSeries",
        "BPName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCityName": null,
        "ThingCountryDescription": null,
    }
}

```

```

        "ThingCountryId": null,
        "ThingAddressRepresentation": null,
        "ThingLocationId": null,
        "EventInfo": "Alert on exceeded temperature",
        "appliication": "exceeded temperature app",
        "temperature": "35",
        "message": "Alert on exceeded temperature"
    }
  ]
}

```

### Example 3

The request `/CompositeEvents/v2/PST_core_automobiles__wheelEPSTs?$filter=Path eq 'wheeltemp'&$expand=HierarchyElements` retrieves the following response in [JSON](#) format:

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/PST_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
          "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/PST_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
          "type": "com.sap.appiot.PST_core_automobiles__wheelEPST"
        },
        "EventId": "2C0108673D43483B94AFEB337646CF9",
        "ExternalId": "exceedtemp-extID",
        "Path": "wheeltemp",
        "BusinessTimestamp": "/Date(1533459600000)/",
        "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
        "EventType": "core.automobiles:exceedTemp",
        "Type": "Danger",
        "Status": "Open",
        "StatusDescription": "Open Status",
        "Severity": 1,
        "SeverityDescription": "High",
        "Code": null,
        "CodeDescription": null,
        "ThingId": "71A184C1345C4DD68D46315181409738",
        "ThingExternalId": "ABCXSeries-extID",
        "ThingProperty": "core.automobiles:ABCXSeries/frwheel",
        "ThingISOCODE": "en",
        "ThingDescription": "ABCXSeries thing description",
        "ThingName": "ABCXSeries",
        "ThingType": "core.automobiles:ABCXSeries",
        "BPName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCityName": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingAddressRepresentation": null,
        "ThingLocationId": null,
        "EventInfo": "Alert on exceeded temperature",
        "appliication": "exceeded temperature app",
        "temperature": "55",
        "message": "Alert on exceeded temperature",
        "HierarchyElements": {
          "results": [
            {

```

```

        "__metadata": {
            "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/
HierarchyElements('2C0108673D43483B94AFEB337646CF9')",
            "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/
HierarchyElements('2C0108673D43483B94AFEB337646CF9')",
            "type": "com.sap.apptot.HierarchyElement"
        },
        "EventId": "2C0108673D43483B94AFEB337646CF9",
        "HierarchyId": "D6B000D0588644571600B10292B22EE6",
        "RootThing": "71A184C1345C4DD68D46315181409738",
        "RootThingType": "core.automobiles:ABCXSeries",
        "RootThingName": "ABCXSeries",
        "RootThingDescription": "ABCXSeries thing
description",
        "RootThingExternalId": "ABCXSeries-extID",
        "HierarchyDescription": "Desccribes hierarchy of
units in the ABCXSeries"
    }
}
},
{
    "__metadata": {
        "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/
PST_core_automobiles__wheelEPSTs('2C9A8540B2A24560AA9CFB2EC209469D')",
        "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/
PST_core_automobiles__wheelEPSTs('2C9A8540B2A24560AA9CFB2EC209469D')",
        "type": "com.sap.apptot.PST_core_automobiles__wheelEPST"
    },
    "EventId": "2C9A8540B2A24560AA9CFB2EC209469D",
    "ExternalId": "exceedtemp-extID",
    "Path": "wheeltemp",
    "BusinessTimestamp": "/Date(1528189200000)/",
    "CorrelationId": "a836f37ef69c4b75a49769bf17ec560f",
    "EventType": "core.automobiles:exceedTemp",
    "Type": "Warning",
    "Status": "Open",
    "StatusDescription": "Open Status",
    "Severity": 2,
    "SeverityDescription": "Medium",
    "Code": null,
    "CodeDescription": null,
    "ThingId": "71A184C1345C4DD68D46315181409738",
    "ThingExternalId": "ABCXSeries-extID",
    "ThingProperty": "core.automobiles:ABCXSeries/frwheel",
    "ThingISOCode": "en",
    "ThingDescription": "ABCXSeries thing description",
    "ThingName": "ABCXSeries",
    "ThingType": "core.automobiles:ABCXSeries",
    "BPName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCityName": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingAddressRepresentation": null,
    "ThingLocationId": null,
    "EventInfo": "Alert on exceeded temperature",
    "application": "exceeded temperature app",
    "temperature": "35",
    "message": "Alert on exceeded temperature",
    "HierarchyElements": {

```

```

        "results": [
            {
                "__metadata": {
                    "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/
HierarchyElements('2C9A8540B2A24560AA9CFB2EC209469D')",
                    "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/v2/
HierarchyElements('2C9A8540B2A24560AA9CFB2EC209469D')",
                    "type": "com.sap.apptot.HierarchyElement"
                },
                "EventId": "2C9A8540B2A24560AA9CFB2EC209469D",
                "HierarchyId": "D6B000D0588644571600B10292B22EE6",
                "RootThing": "71A184C1345C4DD68D46315181409738",
                "RootThingType": "core.automobiles:ABCXSeries",
                "RootThingName": "ABCXSeries",
                "RootThingDescription": "ABCXSeries thing
description",
                "RootThingExternalId": "ABCXSeries-extID",
                "HierarchyDescription": "Desccribes hierarchy of
units in the ABCXSeries"
            }
        ]
    }
}

```

## Related Information

[Composite Events: OData Service \[page 1062\]](#)

### 21.3.5 Read Details of all Events for an Event Type

Retrieves all events for the specified event type

In addition, you can retrieve a subset of all events, according to the filter criteria provided. For a collective request for a set of events, you have these options:

- You can restrict the set of events to be retrieved by filter criteria based on the business timestamp, event type, type, property, and tenant.
- You can define that the set of matching events shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 100 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

URI:

Version	URI
v1	/CompositeEvents/EventType/v1/<event type name>/Events
v2	/CompositeEvents/EventType/v2/<event type name>/Events

Operation Type: CRUD

HTTP Method: [GET](#)

Permissions: <ctodata>.r

### Request Parameters

Parameter	Required	Data Type	Description
event type name	Yes	String	Name of the event type

### Query Request Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; all fields in the result are valid.  To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$filter	No	String	Filter condition to be applied  <b>/v1 version:</b> all fields in the result set are valid. However, the <a href="#">BusinessTimestamp</a> field is not supported.  <b>/v2 version:</b> all fields in the result set are valid, except for the fields <a href="#">BusinessTimestamp</a> , <a href="#">EventType</a> , <a href="#">Type</a> , <a href="#">Property</a> , and <a href="#">Tenant</a> .  The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> .  Multiple filters are supported using the <a href="#">AND</a> separator.

Parameter	Required	Data Type	Description
\$select	No	String	<p>Select condition for the data fields to be included in the result set.</p> <p><b>/v1 version:</b> all fields in the result set are valid.</p> <p><b>/v2 version:</b> all fields in the result set are valid, except for the fields <i>EventType</i>, <i>Type</i>, <i>Property</i>, and <i>Tenant</i>.</p>

## Request Example

/CompositeEvents/EventType/v1/core.automobiles:exceedTemp/Events

## Response

### Response Status and Error Codes

Code	Description
200	Events of a specific event type retrieved successfully

## Payload

Format: *XML*

Media types *JSON* and *XML* are supported. The *JSON* for this method has the structure as depicted in the following examples:

### Example 1

Request /CompositeEvents/EventType/v1/core.automobiles:exceedTemp/Events retrieves the following response:

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v1/core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEBC337646CF9')",
          "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v1/core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEBC337646CF9')",
          "type": "com.sap.apptot.Event"
        },
        "EventId": "2C0108673D43483B94AFEBC337646CF9",
        "ExternalId": "exceedtemp-extID",
        "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
        "BusinessTimestamp": "/Date(1533459600000)/",
        "Status": "Open",

```

```

        "EventType": "core.automobiles:exceedTemp",
        "Type": "Danger",
        "Severity": 1,
        "Source": null,
        "Code": null,
        "ThingId": "71A184C1345C4DD68D46315181409738",
        "Property": "core.automobiles:ABCXSeries/frwheel",
        "Tenant": "com-abc",
        "Description": "Alert on exceeded temperature",
        "DYN_ENT_core_automobiles__wheelEPST": {
            "__deferred": {
                "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v1/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')/
DYN_ENT_core_automobiles__wheelEPST"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v1/
core.automobiles:exceedTemp/Events('2C9A8540B2A24560AA9CFB2EC209469D')",
            "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v1/
core.automobiles:exceedTemp/Events('2C9A8540B2A24560AA9CFB2EC209469D')",
            "type": "com.sap.apptot.Event"
        },
        "EventId": "2C9A8540B2A24560AA9CFB2EC209469D",
        "ExternalId": "exceedtemp-extID",
        "CorrelationId": "a836f37ef69c4b75a49769bf17ec560f",
        "BusinessTimestamp": "/Date(1528189200000)/",
        "Status": "Open",
        "EventType": "core.automobiles:exceedTemp",
        "Type": "Warning",
        "Severity": 2,
        "Source": null,
        "Code": null,
        "ThingId": "71A184C1345C4DD68D46315181409738",
        "Property": "core.automobiles:ABCXSeries/frwheel",
        "Tenant": "com-abc",
        "Description": "Alert on exceeded temperature",
        "DYN_ENT_core_automobiles__wheelEPST": {
            "__deferred": {
                "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v1/
core.automobiles:exceedTemp/Events('2C9A8540B2A24560AA9CFB2EC209469D')/
DYN_ENT_core_automobiles__wheelEPST"
            }
        }
    }
]
}

```

## Example 2

Request `/CompositeEvents/EventType/v2/core.automobiles:exceedTemp/Events` retrieves the following response:

```

{
  "d": {
    "results": [
      {
        "__metadata": {

```

```

        "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
        "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
        "type": "com.sap.apptot.Event"
    },
    "EventId": "2C0108673D43483B94AFEB337646CF9",
    "ExternalId": "exceedtemp-extID",
    "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
    "BusinessTimestamp": "/Date(1533459600000)/",
    "Status": "Open",
    "EventType": "core.automobiles:exceedTemp",
    "Type": "Danger",
    "Severity": 1,
    "Source": null,
    "Code": null,
    "ThingId": "71A184C1345C4DD68D46315181409738",
    "Property": "core.automobiles:ABCXSeries/frwheel",
    "ThingISOCODE": "en",
    "ThingDescription": "ABCXSeries thing description",
    "ThingName": "ABCXSeries",
    "ThingExternalId": "ABCXSeries-extID",
    "ThingType": "core.automobiles:ABCXSeries",
    "BPName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCityName": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingAddressRepresentation": null,
    "ThingLocationId": null,
    "Tenant": "com-abc",
    "DYN_ENT_core_automobiles__wheelEPST": {
        "__deferred": {
            "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')/
DYN_ENT_core_automobiles__wheelEPST"
        }
    }
},
{
    "__metadata": {
        "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C9A8540B2A24560AA9CFB2EC209469D')",
        "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C9A8540B2A24560AA9CFB2EC209469D')",
        "type": "com.sap.apptot.Event"
    },
    "EventId": "2C9A8540B2A24560AA9CFB2EC209469D",
    "ExternalId": "exceedtemp-extID",
    "CorrelationId": "a836f37ef69c4b75a49769bf17ec560f",
    "BusinessTimestamp": "/Date(1528189200000)/",
    "Status": "Open",
    "EventType": "core.automobiles:exceedTemp",
    "Type": "Warning",
    "Severity": 2,
    "Source": null,
    "Code": null,
    "ThingId": "71A184C1345C4DD68D46315181409738",
    "Property": "core.automobiles:ABCXSeries/frwheel",
    "ThingISOCODE": "en",

```



```

        "ThingDescription": "ABCXSeries thing description",
        "ThingName": "ABCXSeries",
        "ThingExternalId": "ABCXSeries-extID",
        "ThingType": "core.automobiles:ABCXSeries",
        "BPName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCityName": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingAddressRepresentation": null,
        "ThingLocationId": null,
        "Tenant": "com-abc",
        "DYN_ENT_core_automobiles__wheelEPST": {
            "__deferred": {
                "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C9A8540B2A24560AA9CFB2EC209469D')/
DYN_ENT_core_automobiles__wheelEPST"
            }
        }
    }
}

```

### Example 3

Request `/CompositeEvents/EventType/v1/core.automobiles:exceedTemp/Events?`  
`$expand=DYN_ENT_core_automobiles__wheelEPST` retrieves the following response:

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
          "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
          "type": "com.sap.apptiot.Event"
        },
        "EventId": "2C0108673D43483B94AFEB337646CF9",
        "ExternalId": "exceedtemp-extID",
        "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
        "BusinessTimestamp": "/Date(1533459600000)/",
        "Status": "Open",
        "EventType": "core.automobiles:exceedTemp",
        "Type": "Danger",
        "Severity": 1,
        "Source": null,
        "Code": null,
        "ThingId": "71A184C1345C4DD68D46315181409738",
        "Property": "core.automobiles:ABCXSeries/frwheel",
        "Tenant": "iotae-citest1",
        "DYN_ENT_core_automobiles__wheelEPST": {
          "__metadata": {
            "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
            "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/

```

```

core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
    "type":
    "com.sap.apptot.DYN_ENT_core_automobiles__wheelEPST"
    },
    "ObjectId": "2C0108673D43483B94AFEB337646CF9",
    "wheeltemp.temperature": "55",
    "wheeltemp.appliction": "exceeded temperature app",
    "wheeltemp.message": "Alert on exceeded temperature",
    "Path": "wheeltemp",
    "Timestamp": "/Date(1536147665246)/"
  }
},
{
  "__metadata": {
    "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C9A8540B2A24560AA9CFB2EC209469D')",
    "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C9A8540B2A24560AA9CFB2EC209469D')",
    "type": "com.sap.apptot.Event"
  },
  "EventId": "2C9A8540B2A24560AA9CFB2EC209469D",
  "ExternalId": "exceedtemp-extID",
  "CorrelationId": "a836f37ef69c4b75a49769bf17ec560f",
  "BusinessTimestamp": "/Date(1528189200000)/",
  "Status": "Open",
  "EventType": "core.automobiles:exceedTemp",
  "Type": "Warning",
  "Severity": 2,
  "Source": null,
  "Code": null,
  "ThingId": "71A184C1345C4DD68D46315181409738",
  "Property": "core.automobiles:ABCXSeries/frwheel",
  "Tenant": "iotae-citest1",
  "DYN_ENT_core_automobiles__wheelEPST": {
    "__metadata": {
      "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C9A8540B2A24560AA9CFB2EC209469D')",
      "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C9A8540B2A24560AA9CFB2EC209469D')",
      "type":
      "com.sap.apptot.DYN_ENT_core_automobiles__wheelEPST"
    },
    "ObjectId": "2C9A8540B2A24560AA9CFB2EC209469D",
    "wheeltemp.temperature": "35",
    "wheeltemp.appliction": "exceeded temperature app",
    "wheeltemp.message": "Alert on exceeded temperature",
    "Path": "wheeltemp",
    "Timestamp": "/Date(1536144468081)/"
  }
}
]
}
}

```

#### Example 4

Request /CompositeEvents/EventType/v2/core.automobiles:exceedTemp/Events?  
 \$expand=DYN\_ENT\_core\_automobiles\_\_wheelEPST retrieves the following response:

```
{
```

```

"d": {
  "results": [
    {
      "__metadata": {
        "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
        "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
        "type": "com.sap.apptot.Event"
      },
      "EventId": "2C0108673D43483B94AFEB337646CF9",
      "ExternalId": "exceedtemp-extID",
      "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
      "BusinessTimestamp": "/Date(1533459600000)/",
      "Status": "Open",
      "EventType": "core.automobiles:exceedTemp",
      "Type": "Danger",
      "Severity": 1,
      "Source": null,
      "Code": null,
      "ThingId": "71A184C1345C4DD68D46315181409738",
      "Property": "core.automobiles:ABCXSeries/frwheel",
      "ThingISOCode": "en",
      "ThingDescription": "ABCXSeries thing description",
      "ThingName": "ABCXSeries",
      "ThingExternalId": "ABCXSeries-extID",
      "ThingType": "core.automobiles:ABCXSeries",
      "BPName": null,
      "BPLandlinePhoneNumber": null,
      "BPMobilePhoneNumber": null,
      "ThingCustomerId": null,
      "ThingRegionDescription": null,
      "ThingCityName": null,
      "ThingCountryDescription": null,
      "ThingCountryId": null,
      "ThingAddressRepresentation": null,
      "ThingLocationId": null,
      "Tenant": "com-abc",
      "DYN_ENT_core_automobiles__wheelEPST": {
        "__metadata": {
          "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
          "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
          "type":
"com.sap.apptot.DYN_ENT_core_automobiles__wheelEPST"
        },
        "ObjectId": "2C0108673D43483B94AFEB337646CF9",
        "wheeltemp.temperature": "55",
        "wheeltemp.appliction": "exceeded temperature app",
        "wheeltemp.message": "Alert on exceeded temperature",
        "Path": "wheeltemp",
        "Timestamp": "/Date(1536147665246)/"
      }
    },
    {
      "__metadata": {
        "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C9A8540B2A24560AA9CFB2EC209469D')",

```

```

        "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C9A8540B2A24560AA9CFB2EC209469D')",
        "type": "com.sap.apptot.Event"
    },
    "EventId": "2C9A8540B2A24560AA9CFB2EC209469D",
    "ExternalId": "exceedtemp-extID",
    "CorrelationId": "a836f37ef69c4b75a49769bfl7ec560f",
    "BusinessTimestamp": "/Date(1528189200000)/",
    "Status": "Open",
    "EventType": "core.automobiles:exceedTemp",
    "Type": "Warning",
    "Severity": 2,
    "Source": null,
    "Code": null,
    "ThingId": "71A184C1345C4DD68D46315181409738",
    "Property": "core.automobiles:ABCXSeries/frwheel",
    "ThingISOCODE": "en",
    "ThingDescription": "ABCXSeries thing description",
    "ThingName": "ABCXSeries",
    "ThingExternalId": "ABCXSeries-extID",
    "ThingType": "core.automobiles:ABCXSeries",
    "BPName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCityName": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingAddressRepresentation": null,
    "ThingLocationId": null,
    "Tenant": "com-abc",
    "DYN_ENT_core_automobiles__wheelEPST": {
        "__metadata": {
            "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C9A8540B2A24560AA9CFB2EC209469D')",
            "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C9A8540B2A24560AA9CFB2EC209469D')",
            "type":
"com.sap.apptot.DYN_ENT_core_automobiles__wheelEPST"
        },
        "ObjectId": "2C9A8540B2A24560AA9CFB2EC209469D",
        "wheeltemp.temperature": "35",
        "wheeltemp.appliction": "exceeded temperature app",
        "wheeltemp.message": "Alert on exceeded temperature",
        "Path": "wheeltemp",
        "Timestamp": "/Date(1536144468081)/"
    }
}
]
}
}

```

## Related Information

[Composite Events: OData Service \[page 1062\]](#)

## 21.3.6 Read Details of an Event for an Event Type

Retrieves details of an event that belongs to a specific event type

### Request

URI:

Version	URI
v1	/CompositeEvents/EventType/v1/<event type name>/Events('<event ID>')
v2	/CompositeEvents/EventType/v2/<event type name>/Events('<event ID>')

Operation Type: CRUD

HTTP Method: [GET](#)

Permissions: <ctodata>.r

### Request Parameters

Parameter	Required	Data Type	Description
event type name	Yes	String	Name of the event type
event ID	Yes	String	Unique identifier of the event

### Request Example

```
/CompositeEvents/EventType/v2/iotae.citest1.core.bbodataeventpkg:TempExceed/  
Events('2C0108673D43483B94AFEBBC337646CF9')
```

### Response

#### Response Status and Error Codes

Code	Description
200	Event details of a specific event type retrieved successfully

### Payload

Format: [XML](#)

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as depicted in the following examples:

### Example 1

Request `/CompositeEvents/EventType/v1/core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')` retrieves the following response:

```
{
  "d": {
    "__metadata": {
      "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v1/core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
      "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v1/core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
      "type": "com.sap.apptot.Event"
    },
    "EventId": "2C0108673D43483B94AFEB337646CF9",
    "ExternalId": "exceedtemp-extID",
    "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
    "BusinessTimestamp": "/Date(1533459600000)/",
    "Status": "Open",
    "EventType": "core.automobiles:exceedTemp",
    "Type": "Danger",
    "Severity": 1,
    "Source": null,
    "Code": null,
    "ThingId": "71A184C1345C4DD68D46315181409738",
    "Property": "core.automobiles:ABCXSeries/frwheel",
    "Tenant": "com-abc",
    "Description": "Alert on exceeded temperature",
    "DYN_ENT_core_automobiles__wheelEPST": {
      "__deferred": {
        "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v1/core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')/DYN_ENT_core_automobiles__wheelEPSTs"
      }
    }
  }
}
```

Request `/CompositeEvents/EventType/v2/core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')` retrieves the following response:

```
{
  "d": {
    "__metadata": {
      "id": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
      "uri": "https://composite-events-odata-ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
      "type": "com.sap.apptot.Event"
    },
    "EventId": "2C0108673D43483B94AFEB337646CF9",
    "ExternalId": "exceedtemp-extID",
    "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
    "BusinessTimestamp": "/Date(1533459600000)/",
    "Status": "Open",
    "EventType": "core.automobiles:exceedTemp",
    "Type": "Danger",
  }
}
```

```

    "Severity": 1,
    "Source": null,
    "Code": null,
    "ThingId": "71A184C1345C4DD68D46315181409738",
    "Property": "core.automobiles:ABCXSeries/frwheel",
    "ThingISOCode": "en",
    "ThingDescription": "ABCXSeries thing description",
    "ThingName": "ABCXSeries",
    "ThingExternalId": "ABCXSeries-extID",
    "ThingType": "core.automobiles:ABCXSeries",
    "BPName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCityName": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingAddressRepresentation": null,
    "ThingLocationId": null,
    "Tenant": "com-abc",
    "DYN_ENT_core_automobiles__wheelEPST": {
      "__deferred": {
        "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEBC337646CF9')/
DYN_ENT_core_automobiles__wheelEPSTs"
      }
    }
  }
}

```

Request /CompositeEvents/EventType/v1/core.automobiles:exceedTemp/  
Events('2C0108673D43483B94AFEBC337646CF9')?

\$expand=DYN\_ENT\_core\_automobiles\_\_wheelEPST retrieves the following response:

```

{
  "d": {
    "__metadata": {
      "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEBC337646CF9')",
      "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEBC337646CF9')",
      "type": "com.sap.apptot.Event"
    },
    "EventId": "2C0108673D43483B94AFEBC337646CF9",
    "ExternalId": "exceedtemp-extID",
    "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
    "BusinessTimestamp": "/Date(1533459600000)/",
    "Status": "Open",
    "EventType": "core.automobiles:exceedTemp",
    "Type": "Danger",
    "Severity": 1,
    "Source": null,
    "Code": null,
    "ThingId": "71A184C1345C4DD68D46315181409738",
    "Property": "core.automobiles:ABCXSeries/frwheel",
    "Description": "Alert on exceeded temperature",
    "Tenant": "com-abc",
    "DYN_ENT_core_automobiles__wheelEPST": {
      "__metadata": {
        "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/

```

```

core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
    "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEB337646CF9')",
    "type": "com.sap.apptot.DYN_ENT_core_automobiles__wheelEPST"
  },
  "ObjectId": "2C0108673D43483B94AFEB337646CF9",
  "wheeltemp.temperature": "55",
  "wheeltemp.appliction": "exceeded temperature app",
  "wheeltemp.message": "Alert on exceeded temperature",
  "Path": "wheeltemp",
  "Timestamp": "/Date(1536147665246)/"
}
}
}

```

Request /CompositeEvents/EventType/v2/core.automobiles:exceedTemp/  
Events('2C0108673D43483B94AFEB337646CF9')?

\$expand=DYN\_ENT\_core\_automobiles\_\_wheelEPST retrieves the following response:

```

{
  "d": {
    "__metadata": {
      "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
      "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/Events('2C0108673D43483B94AFEB337646CF9')",
      "type": "com.sap.apptot.Event"
    },
    "EventId": "2C0108673D43483B94AFEB337646CF9",
    "ExternalId": "exceedtemp-extID",
    "CorrelationId": "946978c7dfe04d7e88bc61fccb47d9ec",
    "BusinessTimestamp": "/Date(1533459600000)/",
    "Status": "Open",
    "EventType": "core.automobiles:exceedTemp",
    "Type": "Danger",
    "Severity": 1,
    "Source": null,
    "Code": null,
    "ThingId": "71A184C1345C4DD68D46315181409738",
    "Property": "core.automobiles:ABCXSeries/frwheel",
    "ThingISOCode": "en",
    "ThingDescription": "ABCXSeries thing description",
    "ThingName": "ABCXSeries",
    "ThingExternalId": "ABCXSeries-extID",
    "ThingType": "core.automobiles:ABCXSeries",
    "BPName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCityName": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingAddressRepresentation": null,
    "ThingLocationId": null,
    "Tenant": "com-abc",
    "DYN_ENT_core_automobiles__wheelEPST": {
      "__metadata": {
        "id": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/

```



```

core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEBC337646CF9')",
    "uri": "https://composite-events-odata-
ci.cfapps.sap.hana.ondemand.com:443/CompositeEvents/EventType/v2/
core.automobiles:exceedTemp/
DYN_ENT_core_automobiles__wheelEPSTs('2C0108673D43483B94AFEBC337646CF9')",
    "type": "com.sap.apptot.DYN_ENT_core_automobiles__wheelEPST"
  },
  "ObjectId": "2C0108673D43483B94AFEBC337646CF9",
  "wheeltemp.temperature": "55",
  "wheeltemp.appliction": "exceeded temperature app",
  "wheeltemp.message": "Alert on exceeded temperature",
  "Path": "wheeltemp",
  "Timestamp": "/Date(1536147665246) /"
}
}
}

```

## 21.4 Composite Thing Configuration for a Specific Thing Hierarchy: OData Service

### Overview

With this method, you can retrieve the thing configuration details for the specified thing hierarchy. The result set includes the following:

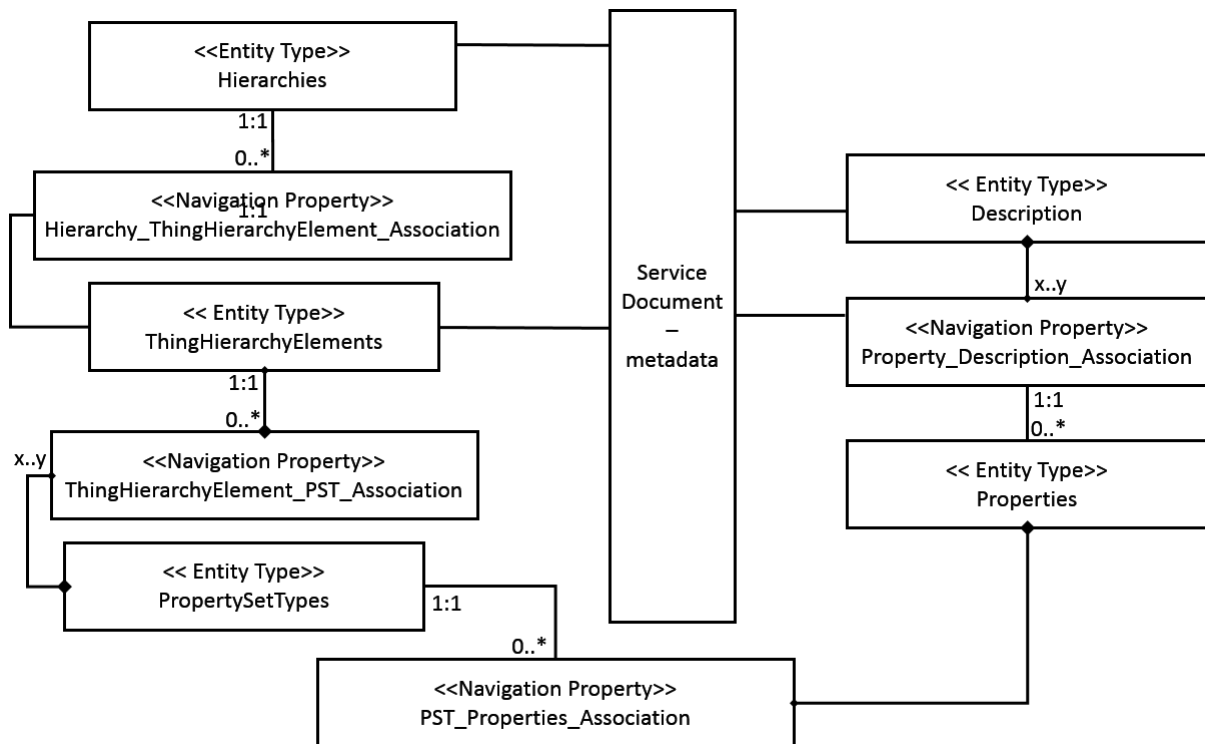
- Details of the specified thing hierarchy.
- All things that belong to the specified thing hierarchy.
- Property set types and properties associated with things that belong to the specified thing hierarchy.

#### OData Version: 2.0

**Root URI:** `http://<server address>[:<port number>]/CompositesThingHierarchy/v1`

**Example for a Root URI in a cloud foundry environment:** `https://cs-hierarchy-meta-sap.cfapps.eu10.hana.ondemand.com/CompositesThingHierarchy`

## Entity Data Model



**Service Metadata URI:** `http://<server address>[:<port number>]/CompositesThingHierarchy/v1/$metadata`

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scope
GET	<a href="#">Read Metadata [page 1095]</a>	/ CompositesThingHierarchy/v1/\$metadata	<ctodata>.r
GET	<a href="#">Read all Hierarchies [page 1098]</a>	/ CompositesThingHierarchy/v1/Hierarchies	<ctodata>.r
GET	<a href="#">Read Details of a Thing Hierarchy [page 1100]</a>	/ CompositesThingHierarchy/v1/Hierarchies('<thing hierarchy ID>')	<ctodata>.r

HTTP Method	Operation	URI	Scope
GET	<a href="#">Read Thing Hierarchy Elements [page 1101]</a>	/CompositesThingHierarchy/v1/Hierarchies('<thing hierarchy ID>')/ThingHierarchyElements	<ctodata>.r
GET	<a href="#">Read Thing Hierarchy Elements and Property Set Types [page 1105]</a>	/CompositesThingHierarchy/v1/Hierarchies('<thing hierarchy ID>')?\$expand=ThingHierarchyElements/PropertySetTypes	<ctodata>.r
GET	<a href="#">Read Thing Hierarchy Elements, Property Set Types, and Properties [page 1109]</a>	/CompositesThingHierarchy/v1/Hierarchies('<thing hierarchy ID>')?\$expand=ThingHierarchyElements/PropertySetTypes/Properties	<ctodata>.r

## 21.4.1 Read Metadata

With this method, you can retrieve metadata of composite things along with thing hierarchy details.

### Request

**URI:** /CompositesThingHierarchy/v1/\$metadata

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <ctodata>.r

### Request Parameters

None

### Request Example

/CompositesThingHierarchy/v1/\$metadata

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

### Format:

Media types supported are [JSON](#) and [XML](#). The [XML](#) for this method has the following structure:

```
<EntityType Name="Hierarchy">
  <Key>
    <PropertyRef Name="HierarchyId"/>
  </Key>
  <Property Name="HierarchyId" Type="Edm.String" Nullable="false"/>
  <Property Name="HierarchyName" Type="Edm.String"/>
  <Property Name="HierarchyExternalId" Type="Edm.String"/>
  <Property Name="ISOCODE" Type="Edm.String"/>
  <Property Name="HierarchyDescription" Type="Edm.String"/>
  <NavigationProperty Name="ThingHierarchyElements"
Relationship="com.sap.apptot.cs.hierarchy.ThingHierarchyElements"
FromRole="Hierarchies" ToRole="ThingHierarchyElements"/>
</EntityType>

<EntityType Name="ThingHierarchyElement">
  <Key>
    <PropertyRef Name="HierarchyId"/>
    <PropertyRef Name="ThingId"/>
  </Key>
  <Property Name="HierarchyId" Type="Edm.String" Nullable="false"/>
  <Property Name="ThingId" Type="Edm.String" Nullable="false"/>
  <Property Name="ThingName" Type="Edm.String"/>
  <Property Name="ISOCODE" Type="Edm.String"/>
  <Property Name="ThingDescription" Type="Edm.String"/>
  <Property Name="ParentThing" Type="Edm.String"/>
  <Property Name="ParentThingName" Type="Edm.String"/>
  <Property Name="ParentThingDescription" Type="Edm.String"/>
  <NavigationProperty Name="PropertySetTypes"
Relationship="com.sap.apptot.cs.hierarchy.PropertySetTypes"
FromRole="ThingHierarchyElements" ToRole="PropertySetTypes"/>
</EntityType>

<EntityType Name="PropertySetType">
  <Key>
    <PropertyRef Name="ThingId"/>
    <PropertyRef Name="ThingType"/>
    <PropertyRef Name="PropertySetId"/>
  </Key>
  <Property Name="ThingId" Type="Edm.String" Nullable="false"/>
  <Property Name="ThingType" Type="Edm.String" Nullable="false"/>
  <Property Name="PropertySetId" Type="Edm.String" Nullable="false"/>
  <Property Name="PropertySetType" Type="Edm.String"/>
  <Property Name="ISOCODE" Type="Edm.String"/>
  <Property Name="PropertySetTypeDescription" Type="Edm.String"/>
  <Property Name="DataCategory" Type="Edm.String"/>
</EntityType>
```

```

        <NavigationProperty Name="Properties"
Relationship="com.sap.appiot.cs.hierarchy.Properties"
FromRole="PropertySetTypes" ToRole="Properties">
        <NavigationProperty Name="Descriptions"
Relationship="com.sap.appiot.cs.hierarchy.Description" FromRole="EventTypes"
ToRole="Descriptions">
</EntityType>

<EntityType Name="Property">
    <Key>
        <PropertyRef Name="PropertyName"/>
        <PropertyRef Name="PropertySetType"/>
    </Key>
    <Property Name="PropertyName" Type="Edm.String" Nullable="false"/>
    <Property Name="PropertyDescription" Type="Edm.String" Nullable="false"/>
    <Property Name="PropertySetType" Type="Edm.String" Nullable="false"/>
    <Property Name="Position" Type="Edm.Int16"/>
    <Property Name="PropertyDataType" Type="Edm.String"/>
    <Property Name="PropertyDataTypeLength" Type="Edm.String"/>
    <Property Name="PropertyHasQualityCode" Type="Edm.String"/>
    <Property Name="UoM" Type="Edm.String"/>
    <Property Name="Annotation" Type="Edm.String"/>
    <NavigationProperty Name="Descriptions"
Relationship="com.sap.appiot.cs.hierarchy.Description" FromRole="Properties"
ToRole="Descriptions">
</EntityType>

<EntityType Name="Description">
    <Property Name="Language" Type="Edm.String"/>
    <Property Name="Description" Type="Edm.String"/>
</EntityType>

<EntityContainer Name="CSThingHierarchy" m:IsDefaultEntityContainer="true">
    <EntitySet Name="Hierarchies"
EntityType="com.sap.appiot.cs.hierarchy.Hierarchy"/>
    <EntitySet Name="ThingHierarchyElements"
EntityType="com.sap.appiot.cs.hierarchy.ThingHierarchyElement"/>
    <EntitySet Name="PropertySetTypes"
EntityType="com.sap.appiot.cs.hierarchy.PropertySetType"/>
    <EntitySet Name="Properties"
EntityType="com.sap.appiot.cs.hierarchy.Property"/>
    <EntitySet Name="Descriptions"
EntityType="com.sap.appiot.cs.hierarchy.Description"/>
</EntityContainer>

```

## Related Information

[Composite Thing Configuration for a Specific Thing Hierarchy: OData Service \[page 1093\]](#)

## 21.4.2 Read all Hierarchies

With this method, you can retrieve details of all hierarchies.

You can retrieve a subset of all hierarchies, according to the filter criteria provided. For a collective request for a set of hierarchies, you have these options:

- You can restrict the set of hierarchies to be retrieved by filter criteria based on the hierarchy ID and hierarchy external ID.
- You can define that the set of matching hierarchies shall not be transmitted all at once, but in a paged manner so that only a small number of results is transmitted, thereby reducing the server load.

By default, the service displays a maximum of 100 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

### Request

**URI:** `/CompositesThingHierarchy/v1/Hierarchies`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<ctodata>.r`

### Request Parameters

None

### Request Example

`/CompositesThingHierarchy/v1/Hierarchies`

### Query String Parameters

Parameter	Required	Data Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set.
<code>\$skip</code>	No	Integer	Number of the first n records to be excluded from the result set.

Parameter	Required	Data Type	Description
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; valid fields are <a href="#">HierarchyId</a> and <a href="#">HierarchyExternalId</a> . To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
\$filter	No	String	Filter condition to be applied; valid fields are <a href="#">HierarchyId</a> and <a href="#">HierarchyExternalId</a> . The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> . Multiple filters are supported using the <a href="#">AND</a> separator. Only valid field is <a href="#">PropertySet Type</a> .
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are <a href="#">HierarchyId</a> and <a href="#">HierarchyExternalId</a> .

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "results": [{
      "__metadata": {
        "id": "https://cs-hierarchy-meta-sap-perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')",
        "uri": "https://cs-hierarchy-meta-sap-perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')",
        "type": "com.sap.applot.hierarchymeta.Hierarchies"
      },
```

```

    "HierarchyId": "E90000901EDA39E31500ED02BBE5CEF0",
    "HierarchyName": "H_Baseplate_000505",
    "HierarchyDescription": "Component Hierarchy for Baseplate_000505",
    "HierarchyExternalId": "H_Baseplate_000505",
    "ISOCODE": "en",
    "ThingHierarchyElements": {
      "__deferred": {
        "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')/ThingHierarchyElements"
      }
    }
  }
}

```

## Related Information

[Composite Thing Configuration for a Specific Thing Hierarchy: OData Service \[page 1093\]](#)

## 21.4.3 Read Details of a Thing Hierarchy

With this method, you can retrieve details of a thing hierarchy with a specific thing hierarchy ID

### Request

**URI:** /CompositesThingHierarchy/v1/Hierarchies('<thing hierarchy ID>')

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <ctodata>.r

### Request Parameters

Parameter	Required	Data Type	Description
HierarchyId	Yes	String	Unique thing hierarchy identifier

### Request Example

/CompositesThingHierarchy/v1/Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')

Retrieves details of the thing hierarchy with the ID E90000901EDA39E31500ED02BBE5CEF0.



## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "__metadata": {
      "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')",
      "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')",
      "type": "com.sap.apptot.hierarchymeta.Hierarchies"
    },
    "HierarchyId": "E90000901EDA39E31500ED02BBE5CEF0",
    "HierarchyName": "H_Baseplate_000505",
    "HierarchyDescription": "Component Hierarchy for Baseplate_000505",
    "HierarchyExternalId": "H_Baseplate_000505",
    "ISOCODE": "en",
    "ThingHierarchyElements": {
      "__deferred": {
        "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')/ThingHierarchyElements"
      }
    }
  }
}
```

## Related Information

[Composite Thing Configuration for a Specific Thing Hierarchy: OData Service \[page 1093\]](#)

## 21.4.4 Read Thing Hierarchy Elements

With this method, you can retrieve thing hierarchy elements for a specific thing hierarchy ID.

By default, the service displays a maximum of 100 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records

retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

## Request

**URI:** `/CompositesThingHierarchy/v1/Hierarchies('<thing hierarchy ID>')/ThingHierarchyElements`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<ctodata>.r`

## Request Parameters

Parameter	Required	Data Type	Description
<code>HierarchyId</code>	Yes	String	Unique thing hierarchy identifier

Parameter	Required	Data Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set. Default value is 20.
<code>\$skip</code>	No	Integer	Number of the first n records to be excluded from the result set. Default value is 20.
<code>\$orderby</code>	No	String	Field name to be used for sorting the result set in ascending order; valid fields are <a href="#">HierarchyId</a> , <a href="#">ThingId</a> , <a href="#">ThingName</a> , <a href="#">ThingType</a> , <a href="#">ParentThing</a> , and <a href="#">ParentThingName</a> . To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.
<code>\$filter</code>	No	String	Filter condition to be applied; valid fields are <a href="#">HierarchyId</a> , <a href="#">ThingId</a> , <a href="#">ThingName</a> , <a href="#">ThingType</a> , <a href="#">ParentThing</a> , and <a href="#">ParentThingName</a> . The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> . Multiple filters are supported using the <a href="#">AND</a> separator. Only valid field is <a href="#">property set type name</a> .

Parameter	Required	Data Type	Description
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are <a href="#">HierarchyId</a> , <a href="#">ThingId</a> , <a href="#">ThingName</a> , <a href="#">ThingType</a> , <a href="#">ParentThing</a> , and <a href="#">ParentThingName</a> .

## Request Example

```
/CompositesThingHierarchy/v1/Hierarchies('E90000901EDA39E31500ED02BBE5CEF0') /
ThingHierarchyElements
```

Retrieves thing hierarchy elements for the specified thing hierarchy ID E90000901EDA39E31500ED02BBE5CEF0.

You can also use the query parameter `$expand` to retrieve thing hierarchy elements for the specified thing hierarchy ID. However, you cannot use query parameters `$top`, `$skip`, `$orderby`, and `$select` along with `$expand`.

```
/CompositesThingHierarchy/v1/Hierarchies('E90000901EDA39E31500ED02BBE5CEF0') ?
$expand=ThingHierarchyElements
```

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as depicted in the following examples:

### Example 1

```
/CompositesThingHierarchy/v1/Hierarchies('E90000901EDA39E31500ED02BBE5CEF0') /
ThingHierarchyElements
```

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
ThingHierarchyElements('E90000901EDA39E31500ED02BBE5CEF0')",
```

```

        "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
ThingHierarchyElements('E90000901EDA39E31500ED02BBE5CEF0')",
        "type": "com.sap.apptot.hierarchymeta.ThingHierarchyElements"
    },
    "HierarchyId": "E90000901EDA39E31500ED02BBE5CEF0",
    "ThingId": "07DA3C0D259242248FDCF15413552D76",
    "ThingName": "Component Hierarchy for Baseplate_000505",
    "ThingDescription": "Baseplate 000505",
    "ParentThing": "",
    "ParentThingName": "",
    "ParentThingDescription": "",
    "PropertySetTypes": {
        "__deferred": {
            "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
ThingHierarchyElements('E90000901EDA39E31500ED02BBE5CEF0')/PropertySetTypes"
        }
    }
}
]
}
}

```

## Example 2

/CompositesThingHierarchy/v1/Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')?  
\$expand=ThingHierarchyElements

```

{
  "d": {
    "__metadata": {
      "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')",
      "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')",
      "type": "com.sap.apptot.hierarchymeta.Hierarchies"
    },
    "HierarchyId": "E90000901EDA39E31500ED02BBE5CEF0",
    "HierarchyName": "H_Baseplate_000505",
    "HierarchyDescription": "Component Hierarchy for Baseplate_000505",
    "HierarchyExternalId": "H_Baseplate_000505",
    "ISOCODE": "en",
    "ThingHierarchyElements": {
      "results": [
        {
          "HierarchyId": "E90000901EDA39E31500ED02BBE5CEF0",
          "ThingId": "07DA3C0D259242248FDCF15413552D76",
          "ThingName": "Baseplate_000505",
          "ThingDescription": "Baseplate 000505",
          "ParentThing": "",
          "ParentThingName": "",
          "ParentThingDescription": ""
        }
      ]
    }
  }
}

```

## Related Information

[Composite Thing Configuration for a Specific Thing Hierarchy: OData Service \[page 1093\]](#)

### 21.4.5 Read Thing Hierarchy Elements and Property Set Types

With this method, you can retrieve the following for the specified thing hierarchy ID:

- Thing hierarchy elements
- Property set types associated with the thing that belongs to the specified thing hierarchy

#### Request

**URI:** `/CompositesThingHierarchy/v1/Hierarchies('<thing hierarchy ID>')?$expand=ThingHierarchyElements/PropertySetTypes`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<ctodata>.r`

#### Request Parameters

Parameter	Required	Data Type	Description
<code>HierarchyId</code>	Yes	String	Unique thing hierarchy identifier

#### Query String Parameters

Parameter	Required	Data Type	Description
<code>\$filter</code>	No	String	Filter condition to be applied; valid fields are <a href="#">HierarchyId</a> , <a href="#">HierarchyExternalId</a> , <a href="#">ThingId</a> , <a href="#">ThingName</a> , <a href="#">ThingType</a> , <a href="#">ParentThing</a> , <a href="#">ParentThingName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> , and <a href="#">DataCategory</a> . The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> . Multiple filters are supported using the <a href="#">AND</a> separator. Only valid field is <a href="#">PropertySetType</a> .

## Request Example

```
/CompositesThingHierarchy/v1/Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')?
$expand=ThingHierarchyElements/PropertySetTypes
```

Retrieves thing hierarchy elements for the specified thing hierarchy ID  
E90000901EDA39E31500ED02BBE5CEF0 along with the property set types.

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "__metadata": {
      "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')",
      "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')",
      "type": "com.sap.apptot.hierarchymeta.Hierarchies"
    },
    "HierarchyId": "E90000901EDA39E31500ED02BBE5CEF0",
    "HierarchyName": "H_Baseplate_000505",
    "HierarchyDescription": "Compönent Hierarchy for Baseplate_000505",
    "HierarchyExternalId": "H_Baseplate_000505",
    "ISOCODE": "en",
    "ThingHierarchyElements": {
      "results": [
        {
          "__metadata": {
            "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
ThingHierarchyElements('E90000901EDA39E31500ED02BBE5CEF0')",
            "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
ThingHierarchyElements('E90000901EDA39E31500ED02BBE5CEF0')",
            "type":
"com.sap.apptot.hierarchymeta.ThingHierarchyElements"
          },
          "HierarchyId": "E90000901EDA39E31500ED02BBE5CEF0",
          "ThingId": "07DA3C0D259242248FDCF15413552D76",
          "ThingName": "Baseplate_000505",
          "ThingDescription": "Baseplate 000505",
          "ParentThing": "",
          "ParentThingName": "",

```

```

        "ParentThingDescription": "",
        "PropertySetTypes": {
            "results": [
                {
                    "ThingId": "07DA3C0D259242248FDCF15413552D76",
                    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
                    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:SVM",
                    "PropertySetType": "svm_score",
                    "ISOCODE": "en",
                    "PropertySetTypeDescription": "Centrifugal Pump
SVM Score for MLE",
                    "DataCategory": "TimeSeriesData"
                },
                {
                    "ThingId": "07DA3C0D259242248FDCF15413552D76",
                    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
                    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:PAM",
                    "PropertySetType": "pam_score",
                    "ISOCODE": "en",
                    "PropertySetTypeDescription": "Centrifugal Pump
PAM Score for MLE",
                    "DataCategory": "TimeSeriesData"
                },
                {
                    "ThingId": "07DA3C0D259242248FDCF15413552D76",
                    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
                    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:MAR",
                    "PropertySetType": "mar_score",
                    "ISOCODE": "en",
                    "PropertySetTypeDescription": "Centrifugal Pump
MAR Score for MLE",
                    "DataCategory": "TimeSeriesData"
                },
                {
                    "ThingId": "07DA3C0D259242248FDCF15413552D76",
                    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
                    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:TEC",
                    "PropertySetType": "tec_score",
                    "ISOCODE": "en",
                    "PropertySetTypeDescription": "Centrifugal Pump
TEC Score for MLE",
                    "DataCategory": "TimeSeriesData"
                },
                {
                    "ThingId": "07DA3C0D259242248FDCF15413552D76",
                    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
                    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:RUL",
                    "PropertySetType": "rul_score",
                    "ISOCODE": "en",
                    "PropertySetTypeDescription": "Centrifugal Pump
RUL Score for MLE",
                    "DataCategory": "TimeSeriesData"
                },
                {
                    "ThingId": "07DA3C0D259242248FDCF15413552D76",
                    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",

```





## 21.4.6 Read Thing Hierarchy Elements, Property Set Types, and Properties

This method retrieves the following for the specified thing hierarchy ID:

- Thing hierarchy elements
- Property set types associated with the thing that belongs to the specified thing hierarchy
- Properties of property set types

### Request

**URI:** /CompositesThingHierarchy/v1/Hierarchies('<thing hierarchy ID>')?  
\$expand=ThingHierarchyElements/PropertySetTypes/Properties

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <ctodata>.r

### Request Parameters

Parameter	Required	Data Type	Description
HierarchyId	Yes	String	Unique thing hierarchy identifier

### Query String Parameters

Parameter	Required	Data Type	Description
\$filter	No	String	Filter condition to be applied; valid fields are <a href="#">HierarchyId</a> , <a href="#">HierarchyExternalId</a> , <a href="#">ThingId</a> , <a href="#">ThingName</a> , <a href="#">ThingType</a> , <a href="#">ParentThing</a> , <a href="#">ParentThingName</a> , <a href="#">PropertySetId</a> , <a href="#">PropertySetType</a> , <a href="#">DataCategory</a> , and <a href="#">PropertyName</a> . The filter operators supported are <a href="#">eq</a> , <a href="#">lt</a> , <a href="#">gt</a> , <a href="#">le</a> , <a href="#">ge</a> , and <a href="#">ne</a> . Multiple filters are supported using the <a href="#">AND</a> separator. Only valid field is <a href="#">PropertySetType</a> .

### Request Example

/CompositesThingHierarchy/v1/Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')?  
\$expand=ThingHierarchyElements/PropertySetTypes/Properties&\$format=json

Retrieves thing hierarchy elements for the specified thing hierarchy ID  
E90000901EDA39E31500ED02BBE5CEF0.

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "__metadata": {
      "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')",
      "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
Hierarchies('E90000901EDA39E31500ED02BBE5CEF0')",
      "type": "com.sap.apptot.hierarchymeta.Hierarchies"
    },
    "HierarchyId": "E90000901EDA39E31500ED02BBE5CEF0",
    "HierarchyName": "H_Baseplate_000505",
    "HierarchyDescription": "Component Hierarchy for Baseplate_000505",
    "HierarchyExternalId": "H_Baseplate_000505",
    "ISOCODE": "en",
    "ThingHierarchyElements": {
      "results": [
        {
          "__metadata": {
            "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
ThingHierarchyElements('E90000901EDA39E31500ED02BBE5CEF0')",
            "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
ThingHierarchyElements('E90000901EDA39E31500ED02BBE5CEF0')",
            "type":
"com.sap.apptot.hierarchymeta.ThingHierarchyElements"
          },
          "HierarchyId": "E90000901EDA39E31500ED02BBE5CEF0",
          "ThingId": "07DA3C0D259242248FDCF15413552D76",
          "ThingName": "Baseplate_000505",
          "ThingDescription": "Baseplate 000505",
          "ParentThing": "",
          "ParentThingName": "",
          "ParentThingDescription": "",
          "PropertySetTypes": {
            "results": [
              {
                "__metadata": {
```

```

        "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3ASVM')",
        "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3ASVM')",
        "type":
"com.sap.apptot.hierarchymeta.PropertySetTypes"
    },
    "ThingId": "07DA3C0D259242248FDCF15413552D76",
    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:SVM",
    "PropertySetType": "svm_score",
    "ISOCODE": "en",
    "PropertySetTypeDescription": "Centrifugal Pump
SVM Score for MLE",
    "DataCategory": "TimeSeriesData",
    "Descriptions": {
        "__deferred": {
            "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3ASVM')/
Descriptions"
        }
    },
    "Properties": {
        "results": [
            {
                "PropertySetType": "svm_score",
                "PropertyName": "score",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",
                "UOM": "",
                "Annotation": "",
                "PropertyDataTypeLength": "",
                "PropertyHasQualityCode": false,
                "ISOCODE": "en",
                "PropertyDescription": "SVM score"
            },
            {
                "PropertySetType": "svm_score",
                "PropertyName": "normalised_score",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",
                "UOM": "",
                "Annotation": "",
                "PropertyDataTypeLength": "",
                "PropertyHasQualityCode": false,
                "ISOCODE": "en",
                "PropertyDescription": "Normalized
SVM score"
            }
        ]
    }
},
{
    "__metadata": {

```

```

        "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3APAM')",
        "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3APAM')",
        "type":
"com.sap.apptot.hierarchymeta.PropertySetTypes"
    },
    "ThingId": "07DA3C0D259242248FDCF15413552D76",
    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:PAM",
    "PropertySetType": "pam_score",
    "ISOCODE": "en",
    "PropertySetTypeDescription": "Centrifugal Pump
PAM Score for MLE",
    "DataCategory": "TimeSeriesData",
    "Descriptions": {
        "__deferred": {
            "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3APAM')/
Descriptions"
        }
    },
    "Properties": {
        "results": [
            {
                "PropertySetType": "pam_score",
                "PropertyName": "score",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",
                "UOM": "",
                "Annotation": "",
                "PropertyDataTypeLength": "",
                "PropertyHasQualityCode": false,
                "ISOCODE": "en",
                "PropertyDescription": "Score"
            },
            {
                "PropertySetType": "pam_score",
                "PropertyName": "PredictedValues",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",
                "UOM": "",
                "Annotation": "",
                "PropertyDataTypeLength": "",
                "PropertyHasQualityCode": false,
                "ISOCODE": "en",
                "PropertyDescription": "Score"
            }
        ]
    }
},
{
    "__metadata": {
        "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/

```

```

PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3AMAR')",
    "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3AMAR')",
    "type":
"com.sap.apiot.hierarchymeta.PropertySetTypes"
    },
    "ThingId": "07DA3C0D259242248FDCF15413552D76",
    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:MAR",
    "PropertySetType": "mar_score",
    "ISOCODE": "en",
    "PropertySetTypeDescription": "Centrifugal Pump
MAR Score for MLE",
    "DataCategory": "TimeSeriesData",
    "Descriptions": {
        "_deferred": {
            "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3AMAR')/
Descriptions"
        }
    },
    "Properties": {
        "results": [
            {
                "PropertySetType": "mar_score",
                "PropertyName": "score",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",
                "UOM": "",
                "Annotation": "",
                "PropertyDataTypeLength": "",
                "PropertyHasQualityCode": false,
                "ISOCODE": "en",
                "PropertyDescription": "MAR Score"
            },
            {
                "PropertySetType": "mar_score",
                "PropertyName": "normalised_score",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",
                "UOM": "",
                "Annotation": "",
                "PropertyDataTypeLength": "",
                "PropertyHasQualityCode": false,
                "ISOCODE": "en",
                "PropertyDescription": "Normalised
MAR Score"
            }
        ]
    },
    {
        "_metadata": {
            "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.

```

```

pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3ATEC')",
    "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3ATEC')",
    "type":
"com.sap.apptot.hierarchymeta.PropertySetTypes"
    },
    "ThingId": "07DA3C0D259242248FDCF15413552D76",
    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:TEC",
    "PropertySetType": "tec_score",
    "ISOCODE": "en",
    "PropertySetTypeDescription": "Centrifugal Pump
TEC Score for MLE",
    "DataCategory": "TimeSeriesData",
    "Descriptions": {
        "__deferred": {
            "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3ATEC')/
Descriptions"
        }
    },
    "Properties": {
        "results": [
            {
                "PropertySetType": "tec_score",
                "PropertyName": "score",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",
                "UOM": "",
                "Annotation": "",
                "PropertyDataTypeLength": "",
                "PropertyHasQualityCode": false,
                "ISOCODE": "en",
                "PropertyDescription": "TEC Score"
            },
            {
                "PropertySetType": "tec_score",
                "PropertyName": "predicted_class",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",
                "UOM": "",
                "Annotation": "",
                "PropertyDataTypeLength": "",
                "PropertyHasQualityCode": false,
                "ISOCODE": "en",
                "PropertyDescription": "TEC
Predicted Class"
            }
        ]
    }
},
{
    "__metadata": {
        "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.

```

```

pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3ARUL')",
    "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3ARUL')",
    "type":
"com.sap.apptot.hierarchymeta.PropertySetTypes"
    },
    "ThingId": "07DA3C0D259242248FDCF15413552D76",
    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:RUL",
    "PropertySetType": "rul_score",
    "ISOCODE": "en",
    "PropertySetTypeDescription": "Centrifugal Pump
RUL Score for MLE",
    "DataCategory": "TimeSeriesData",
    "Descriptions": {
        "__deferred": {
            "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3ARUL')/
Descriptions"
        }
    },
    "Properties": {
        "results": [
            {
                "PropertySetType": "rul_score",
                "PropertyName": "survival_curve",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",
                "UOM": "",
                "Annotation": "",
                "PropertyDataTypeLength": "",
                "PropertyHasQualityCode": false,
                "ISOCODE": "en",
                "PropertyDescription": "survival
curve"
            },
            {
                "PropertySetType": "rul_score",
                "PropertyName":
"remaining_useful_life",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",
                "UOM": "",
                "Annotation": "",
                "PropertyDataTypeLength": "",
                "PropertyHasQualityCode": false,
                "ISOCODE": "en",
                "PropertyDescription": "remaining
useful life"
            },
            {
                "PropertySetType": "rul_score",
                "PropertyName":
"failure_probability",
                "Position": "0",
                "PropertyDataType":
"NumericFlexible",

```

```

    "UOM": "",
    "Annotation": "",
    "PropertyDataTypeLength": "",
    "PropertyHasQualityCode": false,
    "ISOCODE": "en",
    "PropertyDescription": "probability
of failure"
  },
  {
    "PropertySetType": "rul_score",
    "PropertyName": "age",
    "Position": "0",
    "PropertyDataType":
    "UOM": "",
    "Annotation": "",
    "PropertyDataTypeLength": "",
    "PropertyHasQualityCode": false,
    "ISOCODE": "en",
    "PropertyDescription": "EMD Score"
  }
]
},
{
  "__metadata": {
    "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3AEMD')",
    "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3AEMD')",
    "type":
    "com.sap.apptot.hierarchymeta.PropertySetTypes"
  },
  "ThingId": "07DA3C0D259242248FDCF15413552D76",
  "ThingType":
    "pdms.test.pdms.assets.pumps:Baseplate",
    "PropertySetId":
    "pdms.test.com.sap.pdms.datascience.scoring:EMD",
    "PropertySetType": "emd_score",
    "ISOCODE": "en",
    "PropertySetTypeDescription": "Centrifugal Pump
EMD Score for MLE",
    "DataCategory": "TimeSeriesData",
    "Descriptions": {
      "__deferred": {
        "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3AEMD')/
Descriptions"
      }
    },
    "Properties": {
      "results": [
        {
          "PropertySetType": "emd_score",
          "PropertyName": "normalised_score",
          "Position": "0",
          "PropertyDataType":
          "UOM": "",

```



```

        "Annotation": "",
        "PropertyDataTypeLength": "",
        "PropertyHasQualityCode": false,
        "ISOCODE": "en",
        "PropertyDescription": "Normalised
EMD Score"
    },
    {
        "PropertySetType": "emd_score",
        "PropertyName": "reading_count",
        "Position": "0",
        "PropertyDataType":
"NumericFlexible",
        "UOM": "",
        "Annotation": "",
        "PropertyDataTypeLength": "",
        "PropertyHasQualityCode": false,
        "ISOCODE": "en",
        "PropertyDescription": "Number of
readings contributing to score"
    },
    {
        "PropertySetType": "emd_score",
        "PropertyName": "score",
        "Position": "0",
        "PropertyDataType":
"NumericFlexible",
        "UOM": "",
        "Annotation": "",
        "PropertyDataTypeLength": "",
        "PropertyHasQualityCode": false,
        "ISOCODE": "en",
        "PropertyDescription": "EMD Score"
    }
]
},
{
    "_metadata": {
        "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3APCA')",
        "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps
%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3APCA')",
        "type":
"com.sap.apptot.hierarchymeta.PropertySetTypes"
    },
    "ThingId": "07DA3C0D259242248FDCF15413552D76",
    "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
    "PropertySetId":
"pdms.test.com.sap.pdms.datascience.scoring:PCA",
    "PropertySetType": "pca_score",
    "ISOCODE": "en",
    "PropertySetTypeDescription": "Centrifugal Pump
PCA Score for MLE",
    "DataCategory": "TimeSeriesData",
    "Descriptions": {
        "_deferred": {
            "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps

```

```

%3ABaseplate',PropertySetId='pdms.test.com.sap.pdms.datascience.scoring%3APCA')/
Descriptions"
    }
  },
  "Properties": {
    "results": [
      {
        "PropertySetType": "pca_score",
        "PropertyName": "score",
        "Position": "0",
        "PropertyDataType":
"NumericFlexible",
        "UOM": "",
        "Annotation": "",
        "PropertyDataTypeLength": "",
        "PropertyHasQualityCode": false,
        "ISOCODE": "en",
        "PropertyDescription": "PCA Score"
      },
      {
        "PropertySetType": "pca_score",
        "PropertyName": "normalised_score",
        "Position": "0",
        "PropertyDataType":
"NumericFlexible",
        "UOM": "",
        "Annotation": "",
        "PropertyDataTypeLength": "",
        "PropertyHasQualityCode": false,
        "ISOCODE": "en",
        "PropertyDescription": "Normalised
PCA Score"
      }
    ]
  }
},
{
  "__metadata": {
    "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps%3ABaseplate',PropertySetId='pdms.test.pdms.assets.pumps
%3AMLEPumpProperties')",
    "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps%3ABaseplate',PropertySetId='pdms.test.pdms.assets.pumps
%3AMLEPumpProperties')",
    "type":
"com.sap.apptot.hierarchymeta.PropertySetTypes"
  },
  "ThingId": "07DA3C0D259242248FDCF15413552D76",
  "ThingType":
"pdms.test.pdms.assets.pumps:Baseplate",
  "PropertySetId":
"pdms.test.pdms.assets.pumps:MLEPumpProperties",
  "PropertySetType": "mlePumpProperties",
  "ISOCODE": "en",
  "PropertySetTypeDescription": "Pump Properties
for MLE",
  "DataCategory": "TimeSeriesData",
  "Descriptions": {
    "__deferred": {
      "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps%3ABaseplate',PropertySetId='pdms.test.pdms.assets.pumps
%3AMLEPumpProperties')/Descriptions"
    }
  }
}
}
}

```

```

    },
    "Properties": {
      "results": [
        {
          "PropertySetType":
            "mlePumpProperties",
            "PropertyName": "FAILURES",
            "Position": "0",
            "PropertyDataType": "Numeric",
            "UOM": "",
            "Annotation": "",
            "PropertyDataTypeLength": "",
            "PropertyHasQualityCode": false,
            "ISOCODE": "en",
            "PropertyDescription": "Failures in
0 or 1"
        },
        {
          "PropertySetType":
            "mlePumpProperties",
            "PropertyName": "PRESSURE",
            "Position": "0",
            "PropertyDataType":
            "NumericFlexible",
            "UOM": "Bar",
            "Annotation": "",
            "PropertyDataTypeLength": "",
            "PropertyHasQualityCode": false,
            "ISOCODE": "en",
            "PropertyDescription": "Pressue"
        },
        {
          "PropertySetType":
            "mlePumpProperties",
            "PropertyName": "TEMPERATURE",
            "Position": "0",
            "PropertyDataType":
            "NumericFlexible",
            "UOM": "°C",
            "Annotation": "",
            "PropertyDataTypeLength": "",
            "PropertyHasQualityCode": false,
            "ISOCODE": "en",
            "PropertyDescription": "Temperature"
        }
      ]
    }
  },
  {
    "_metadata": {
      "id": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76', ThingType='pdms.test.
pdms.assets.pumps%3ABaseplate', PropertySetId='pdms.test.pdms.assets.pumps
%3AHealth')",
      "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76', ThingType='pdms.test.
pdms.assets.pumps%3ABaseplate', PropertySetId='pdms.test.pdms.assets.pumps
%3AHealth')",
      "type":
        "com.sap.apiot.hierarchymeta.PropertySetTypes"
    },
    "ThingId": "07DA3C0D259242248FDCF15413552D76",
    "ThingType":
      "pdms.test.pdms.assets.pumps:Baseplate",

```

```

    "PropertySetId":
      "pdms.test.pdms.assets.pumps:Health",
      "PropertySetType": "healthState",
      "ISOCODE": "en",
      "PropertySetTypeDescription": "Baseplate Health
State",
      "DataCategory": "TimeSeriesData",
      "Descriptions": {
        "__deferred": {
          "uri": "https://cs-hierarchy-meta-sap-
perf.cfapps.sap.hana.ondemand.com:443/CompositesThingHierarchy/v1/
PropertySetTypes(ThingId='07DA3C0D259242248FDCF15413552D76',ThingType='pdms.test.
pdms.assets.pumps%3ABaseplate',PropertySetId='pdms.test.pdms.assets.pumps
%3AHealth')/Descriptions"
        }
      },
      "Properties": {
        "results": [
          {
            "PropertySetType": "healthState",
            "PropertyName": "SCORE",
            "Position": "0",
            "PropertyDataType":
              "NumericFlexible",
            "UOM": "%",
            "Annotation":
              "pdms.test.pdms.assets.pumps:Discrete",
            "PropertyDataTypeLength": "",
            "PropertyHasQualityCode": false,
            "ISOCODE": "en",
            "PropertyDescription": "Raw Health
Score"
          },
          {
            "PropertySetType": "healthState",
            "PropertyName": "STATUS",
            "Position": "0",
            "PropertyDataType":
              "NumericFlexible",
            "UOM": "",
            "Annotation": "",
            "PropertyDataTypeLength": "",
            "PropertyHasQualityCode": false,
            "ISOCODE": "en",
            "PropertyDescription": "Health
Status"
          }
        ]
      }
    }
  }
}

```

## Related Information

Composite Thing Configuration for a Specific Thing Hierarchy: OData Service [page 1093]

## 21.5 Thing Detail

Service retrieves details of the specified thing

### Overview

The service retrieves thing entity details and snapshot data across all property set types that has usage within the specified thing type. In addition, the service retrieves reference property details.

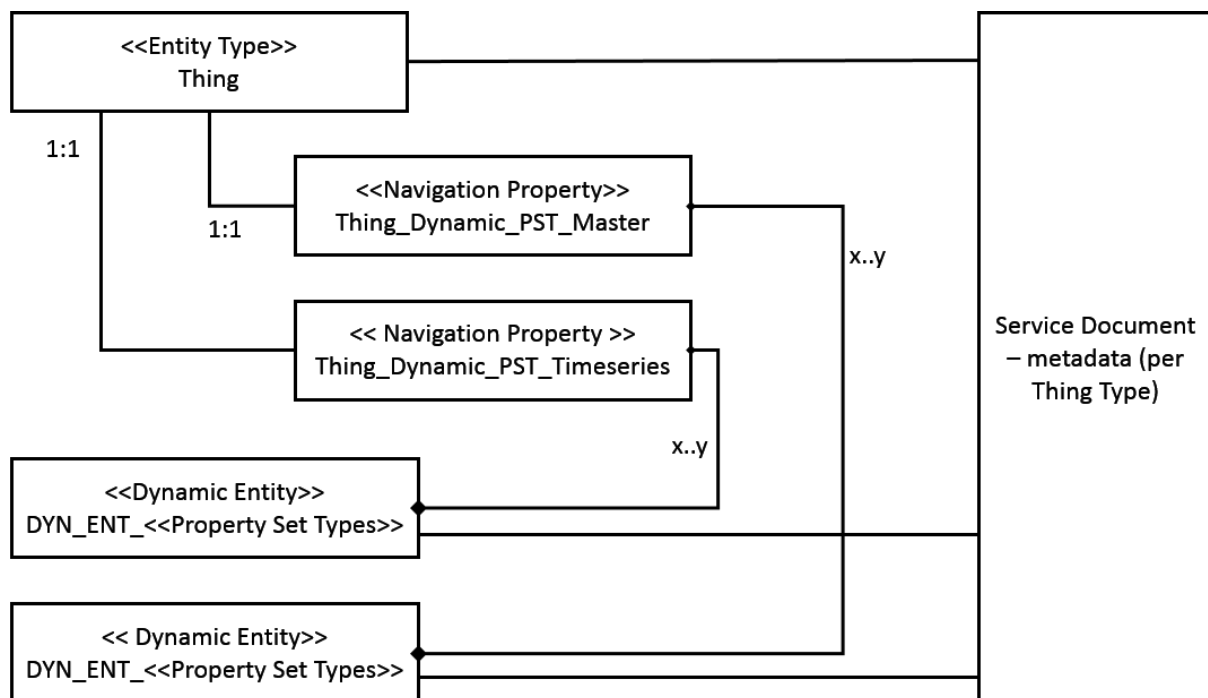
**OData Version: 2.0**

**Root URI:** `http://<server address>[:<port number>]/CompositeThings/ThingType/v1/<thing type name>`

**Example for a base URI in a cloud foundry environment:** `https://details-thing-sap.cfapps.eu10.hana.ondemand.com/CompositeThings/v1/<thing type name>`

### Entity Data Model

The following diagram illustrates the EDM of Thing Detail OData services:



## Related Information

[Read Metadata of Thing Detail \[page 1122\]](#)

[Read Details of a Thing \[page 1129\]](#)

[Read Property Set Data for a Thing \[page 1132\]](#)

## 21.5.1 Read Metadata of Thing Detail

Retrieves metadata of thing detail OData service

The metadata contains an annotation for the dynamic property set type entity. The annotation `sap:semantics="timeseries"` indicates that the property set type belongs to the data category *TimeSeriesData*. Any property set type without an annotation indicates that it belongs to the data category *MasterData*.

### Request

**URI:** `/CompositeThings/ThingType/v1/<thing type name>/$metadata`

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** `<thngdtl.r>`

### Request Header Parameters

Header	Required	Description
<code>sap-iot-issync</code>	No	Indicates whether the service retrieves the latest data creation time for the specified thing ID.  Permissible values are <b>true</b> or <b>false</b>

### Request Parameters

Parameter	Required	Data Type	Description
<code>thing type name</code>	Yes	String	Name of the thing type

### Request Example

`/CompositeThings/ThingType/v1/test.sandbox.devxperf.package:BMW6Series/$metadata`

## Response

### Response Status and Error Codes

Code	Description
200	Metadata retrieved successfully

## Payload

The following response payload contains the property `SyncedTimestamp` only when the request header parameter `sap-iot-issync` is set **true**.

```
URI:<EntityType Name="DYN_ENT_core_automobiles__Engine"
sap:semantics="timeseries">
  <Key>
    <PropertyRef Name="ThingId"></PropertyRef>
  </Key>
  <Property Name="ThingId" Type="Edm.String" Nullable="false"></
Property>
  <Property Name="engine.RPM" Type="Edm.Int32"></Property>
  <Property Name="engine.RPM.Timestamp"
Type="Edm.DateTimeOffset"></Property>
  <Property Name="engine.Temperature" Type="Edm.Int32"></Property>
  <Property Name="engine.Temperature.Timestamp"
Type="Edm.DateTimeOffset"></Property>
</EntityType>
<EntityType Name="DYN_ENT_core_automobiles__Model">
  <Key>
    <PropertyRef Name="ThingId"></PropertyRef>
  </Key>
  <Property Name="ThingId" Type="Edm.String" Nullable="false"></
Property>
  <Property Name="modelData.Color" Type="Edm.String"></Property>
  <Property Name="modelData.Color.Timestamp"
Type="Edm.DateTimeOffset"></Property>
  <Property Name="modelData.Make" Type="Edm.String"></Property>
  <Property Name="modelData.Make.Timestamp"
Type="Edm.DateTimeOffset"></Property>
  <Property Name="modelData.Registration" Type="Edm.String"></
Property>
  <Property Name="modelData.Registration.Timestamp"
Type="Edm.DateTimeOffset"></Property>
</EntityType>
<EntityType Name="DYN_ENT_core_automobiles__LocationMaster">
  <Key>
    <PropertyRef Name="ThingId"></PropertyRef>
  </Key>
  <Property Name="ThingId" Type="Edm.String" Nullable="false"></
Property>
  <Property Name="locationData.Country" Type="Edm.String"></
Property>
  <Property Name="locationData.Country.Timestamp"
Type="Edm.DateTimeOffset"></Property>
  <Property Name="locationData.OriginPlace" Type="Edm.String"></
Property>
  <Property Name="locationData.OriginPlace.Timestamp"
Type="Edm.DateTimeOffset"></Property>
</EntityType>
<EntityType Name="Thing">
```

```

        <Key>
          <PropertyRef Name="ThingId"></PropertyRef>
        </Key>
        <Property Name="ThingId" Type="Edm.String" Nullable="false">
          <Documentation>
            <LongDescription>Thing unique ID</LongDescription>
          </Documentation>
        </Property>
        <Property Name="ThingISOCODE" Type="Edm.String">
          <Documentation>
            <LongDescription>Thing Language code</LongDescription>
          </Documentation>
        </Property>
        <Property Name="ThingDescription" Type="Edm.String">
          <Documentation>
            <LongDescription>Thing description</LongDescription>
          </Documentation>
        </Property>
        <Property Name="ThingName" Type="Edm.String" Nullable="false">
          <Documentation>
            <LongDescription>Thing name</LongDescription>
          </Documentation>
        </Property>
        <Property Name="ThingExternalId" Type="Edm.String">
          <Documentation>
            <LongDescription>Thing external id</LongDescription>
          </Documentation>
        </Property>
        <Property Name="ThingType" Type="Edm.String">
          <Documentation>
            <LongDescription>Thing Type ID associated with the
Thing</LongDescription>
          </Documentation>
        </Property>
        <Property Name="ObjectGroup" Type="Edm.String">
          <Documentation>
            <LongDescription>Object group used to create the Thing</
LongDescription>
          </Documentation>
        </Property>
        <Property Name="BPName" Type="Edm.String">
          <Documentation>
            <LongDescription>Business partner name associated for
the Thing</LongDescription>
          </Documentation>
        </Property>
        <Property Name="BPLandlinePhoneNumber" Type="Edm.String">
          <Documentation>
            <LongDescription>Landline phone number of Business
partner</LongDescription>
          </Documentation>
        </Property>
        <Property Name="BPMobilePhoneNumber" Type="Edm.String">
          <Documentation>
            <LongDescription>Mobile phone number of Business
partner</LongDescription>
          </Documentation>
        </Property>
        <Property Name="ThingCustomerId" Type="Edm.String">
          <Documentation>
            <LongDescription>Business partner ID associated for the
Thing</LongDescription>
          </Documentation>
        </Property>
        <Property Name="ThingRegionDescription" Type="Edm.String">
          <Documentation>
            <LongDescription>Region where Thing is located</
LongDescription>
          </Documentation>
        </Property>

```



```

        </Documentation>
    </Property>
    <Property Name="ThingCountryDescription" Type="Edm.String">
        <Documentation>
            <LongDescription>Country where the Thing is located</
LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCountryId" Type="Edm.String">
        <Documentation>
            <LongDescription>Country code where the Thing is
located</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCommunicationData" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing address details</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingLocationId" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing location ID</LongDescription>
        </Documentation>
    </Property>
    <Property Name="AlternateId" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing alternate ID</LongDescription>
        </Documentation>
    </Property>
    <Property Name="SyncedTimestamp" Type="Edm.String">
        <Documentation>
            <LongDescription>Synced TimeStamp</LongDescription>
        </Documentation>
    </Property>
    <NavigationProperty Name="DYN_ENT_core_automobiles__Model"
Relationship="com.sap.apptot.DYN_ENT_core_automobiles__Model"
FromRole="ThingEntities" ToRole="DYN_ENT_core_automobiles__ModelEntities"></
NavigationProperty>
    <NavigationProperty Name="DYN_ENT_core_automobiles__Wheel"
Relationship="com.sap.apptot.DYN_ENT_core_automobiles__Wheel"
FromRole="ThingEntities" ToRole="DYN_ENT_core_automobiles__WheelEntities"></
NavigationProperty>
    <NavigationProperty Name="DYN_ENT_core_automobiles__Engine"
Relationship="com.sap.apptot.DYN_ENT_core_automobiles__Engine"
FromRole="ThingEntities" ToRole="DYN_ENT_core_automobiles__EngineEntities"></
NavigationProperty>
    <NavigationProperty Name="DYN_ENT_core_automobiles__DriveUnit"
Relationship="com.sap.apptot.DYN_ENT_core_automobiles__DriveUnit"
FromRole="ThingEntities" ToRole="DYN_ENT_core_automobiles__DriveUnitEntities"></
NavigationProperty>
    <NavigationProperty
Name="DYN_ENT_core_automobiles__LocationMaster"
Relationship="com.sap.apptot.DYN_ENT_core_automobiles__LocationMaster"
FromRole="ThingEntities"
ToRole="DYN_ENT_core_automobiles__LocationMasterEntities"></NavigationProperty>
    </EntityType>
    <EntityType Name="DYN_ENT_core_automobiles__DriveUnit"
sap:semantics="timeseries">
        <Key>
            <PropertyRef Name="ThingId"></PropertyRef>
        </Key>
        <Property Name="ThingId" Type="Edm.String" Nullable="false"></
Property>
        <Property Name="leftDriveUnit.Acceleration" Type="Edm.Int32"></
Property>
        <Property Name="leftDriveUnit.Acceleration.Timestamp"
Type="Edm.DateTimeOffset"></Property>

```

```

        <Property Name="leftDriveUnit.Rotation" Type="Edm.Int32"></
Property>
        <Property Name="leftDriveUnit.Rotation.Timestamp"
Type="Edm.DateTimeOffset"></Property>
    </EntityType>
    <EntityType Name="DYN_ENT_core_automobiles__Wheel"
sap:semantics="timeseries">
        <Key>
            <PropertyRef Name="ThingId"></PropertyRef>
        </Key>
        <Property Name="ThingId" Type="Edm.String" Nullable="false"></
Property>
        <Property Name="flWheel.PSI" Type="Edm.Int32"></Property>
        <Property Name="flWheel.PSI.PressureUnitOfMeasure"
Type="Edm.String" AttributeType="com.sap.iot.core.UoM"></Property>
        <Property Name="flWheel.PSI.PressureLowerThreshold"
Type="Edm.Int32" AttributeType="com.sap.iot.core.LowerThreshold"></Property>
        <Property Name="flWheel.PSI.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="flWheel.PSI.PressureLowerThreshold.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="flWheel.PSI.PressureUnitOfMeasure.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="flWheel.Rotation" Type="Edm.Int32"></Property>
        <Property Name="flWheel.Rotation.RotationUnitOfMeasure"
Type="Edm.String" AttributeType="com.sap.iot.core.UoM"></Property>
        <Property Name="flWheel.Rotation.RotationLowerThreshold"
Type="Edm.Int32" AttributeType="com.sap.iot.core.LowerThreshold"></Property>
        <Property Name="flWheel.Rotation.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property
Name="flWheel.Rotation.RotationLowerThreshold.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property
Name="flWheel.Rotation.RotationUnitOfMeasure.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="frWheel.PSI" Type="Edm.Int32"></Property>
        <Property Name="frWheel.PSI.PressureUnitOfMeasure"
Type="Edm.String" AttributeType="com.sap.iot.core.UoM"></Property>
        <Property Name="frWheel.PSI.PressureLowerThreshold"
Type="Edm.Int32" AttributeType="com.sap.iot.core.LowerThreshold"></Property>
        <Property Name="frWheel.PSI.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="frWheel.PSI.PressureUnitOfMeasure.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="frWheel.PSI.PressureLowerThreshold.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="frWheel.Rotation" Type="Edm.Int32"></Property>
        <Property Name="frWheel.Rotation.RotationUnitOfMeasure"
Type="Edm.String" AttributeType="com.sap.iot.core.UoM"></Property>
        <Property Name="frWheel.Rotation.RotationLowerThreshold"
Type="Edm.Int32" AttributeType="com.sap.iot.core.LowerThreshold"></Property>
        <Property Name="frWheel.Rotation.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property
Name="frWheel.Rotation.RotationUnitOfMeasure.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property
Name="frWheel.Rotation.RotationLowerThreshold.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="rlWheel.PSI" Type="Edm.Int32"></Property>
        <Property Name="rlWheel.PSI.PressureUnitOfMeasure"
Type="Edm.String" AttributeType="com.sap.iot.core.UoM"></Property>
        <Property Name="rlWheel.PSI.PressureLowerThreshold"
Type="Edm.Int32" AttributeType="com.sap.iot.core.LowerThreshold"></Property>
        <Property Name="rlWheel.PSI.Timestamp"
Type="Edm.DateTimeOffset"></Property>

```

```

        <Property Name="rlWheel.PSI.PressureUnitOfMeasure.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="rlWheel.PSI.PressureLowerThreshold.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="rlWheel.Rotation" Type="Edm.Int32"></Property>
        <Property Name="rlWheel.Rotation.RotationLowerThreshold"
Type="Edm.Int32" AttributeType="com.sap.iot.core.LowerThreshold"></Property>
        <Property Name="rlWheel.Rotation.RotationUnitOfMeasure"
Type="Edm.String" AttributeType="com.sap.iot.core.UoM"></Property>
        <Property Name="rlWheel.Rotation.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property
Name="rlWheel.Rotation.RotationUnitOfMeasure.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property
Name="rlWheel.Rotation.RotationLowerThreshold.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="rrWheel.PSI" Type="Edm.Int32"></Property>
        <Property Name="rrWheel.PSI.PressureUnitOfMeasure"
Type="Edm.String" AttributeType="com.sap.iot.core.UoM"></Property>
        <Property Name="rrWheel.PSI.PressureLowerThreshold"
Type="Edm.Int32" AttributeType="com.sap.iot.core.LowerThreshold"></Property>
        <Property Name="rrWheel.PSI.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="rrWheel.PSI.PressureLowerThreshold.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="rrWheel.PSI.PressureUnitOfMeasure.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="rrWheel.Rotation" Type="Edm.Int32"></Property>
        <Property Name="rrWheel.Rotation.RotationUnitOfMeasure"
Type="Edm.String" AttributeType="com.sap.iot.core.UoM"></Property>
        <Property Name="rrWheel.Rotation.RotationLowerThreshold"
Type="Edm.Int32" AttributeType="com.sap.iot.core.LowerThreshold"></Property>
        <Property Name="rrWheel.Rotation.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property
Name="rrWheel.Rotation.RotationLowerThreshold.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property
Name="rrWheel.Rotation.RotationUnitOfMeasure.Timestamp"
Type="Edm.DateTimeOffset"></Property>
    </EntityType>
    <Association Name="DYN_ENT_core_automobiles_Model">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.DYN_ENT_core_automobiles_Model"
Multiplicity="1" Role="DYN_ENT_core_automobiles__ModelEntities"></End>
    </Association>
    <Association Name="DYN_ENT_core_automobiles_Wheel">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.DYN_ENT_core_automobiles_Wheel"
Multiplicity="1" Role="DYN_ENT_core_automobiles__WheelEntities"></End>
    </Association>
    <Association Name="DYN_ENT_core_automobiles_Engine">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.DYN_ENT_core_automobiles_Engine"
Multiplicity="1" Role="DYN_ENT_core_automobiles__EngineEntities"></End>
    </Association>
    <Association Name="DYN_ENT_core_automobiles_DriveUnit">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.DYN_ENT_core_automobiles_DriveUnit"
Multiplicity="1" Role="DYN_ENT_core_automobiles__DriveUnitEntities"></End>
    </Association>
    <Association Name="DYN_ENT_core_automobiles_LocationMaster">

```

```

        <End Type="com.sap.appiot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
    <End
Type="com.sap.appiot.DYN_ENT_core_automobiles_LocationMaster" Multiplicity="1"
Role="DYN_ENT_core_automobiles_LocationMasterEntities"></End>
    </Association>
    <EntityContainer Name="CompositeThingsThingType"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="DYN_ENT_core_automobiles_Engines"
EntityType="com.sap.appiot.DYN_ENT_core_automobiles_Engine"></EntitySet>
        <EntitySet Name="DYN_ENT_core_automobiles_Models"
EntityType="com.sap.appiot.DYN_ENT_core_automobiles_Model"></EntitySet>
        <EntitySet Name="DYN_ENT_core_automobiles_LocationMasters"
EntityType="com.sap.appiot.DYN_ENT_core_automobiles_LocationMaster"></EntitySet>
        <EntitySet Name="Things" EntityType="com.sap.appiot.Thing"></
EntitySet>
        <EntitySet Name="DYN_ENT_core_automobiles_DriveUnits"
EntityType="com.sap.appiot.DYN_ENT_core_automobiles_DriveUnit"></EntitySet>
        <EntitySet Name="DYN_ENT_core_automobiles_Wheels"
EntityType="com.sap.appiot.DYN_ENT_core_automobiles_Wheel"></EntitySet>
        <AssociationSet Name="DYN_ENT_core_automobiles_Model"
Association="com.sap.appiot.DYN_ENT_core_automobiles_Model">
            <End EntitySet="Things" Role="ThingEntities"></End>
            <End EntitySet="DYN_ENT_core_automobiles_Models"
Role="DYN_ENT_core_automobiles_ModelEntities"></End>
        </AssociationSet>
        <AssociationSet Name="DYN_ENT_core_automobiles_Wheel"
Association="com.sap.appiot.DYN_ENT_core_automobiles_Wheel">
            <End EntitySet="Things" Role="ThingEntities"></End>
            <End EntitySet="DYN_ENT_core_automobiles_Wheels"
Role="DYN_ENT_core_automobiles_WheelEntities"></End>
        </AssociationSet>
        <AssociationSet Name="DYN_ENT_core_automobiles_Engine"
Association="com.sap.appiot.DYN_ENT_core_automobiles_Engine">
            <End EntitySet="Things" Role="ThingEntities"></End>
            <End EntitySet="DYN_ENT_core_automobiles_Engines"
Role="DYN_ENT_core_automobiles_EngineEntities"></End>
        </AssociationSet>
        <AssociationSet Name="DYN_ENT_core_automobiles_DriveUnit"
Association="com.sap.appiot.DYN_ENT_core_automobiles_DriveUnit">
            <End EntitySet="Things" Role="ThingEntities"></End>
            <End EntitySet="DYN_ENT_core_automobiles_DriveUnits"
Role="DYN_ENT_core_automobiles_DriveUnitEntities"></End>
        </AssociationSet>
        <AssociationSet Name="DYN_ENT_core_automobiles_LocationMaster"
Association="com.sap.appiot.DYN_ENT_core_automobiles_LocationMaster">
            <End EntitySet="Things" Role="ThingEntities"></End>
            <End EntitySet="DYN_ENT_core_automobiles_LocationMasters"
Role="DYN_ENT_core_automobiles_LocationMasterEntities"></End>
        </AssociationSet>
    </EntityContainer>

```

## Related Information

[Thing Detail \[page 1121\]](#)

## 21.5.2 Read Details of a Thing

Retrieves details of the specified thing

### Request

**URI:** /CompositeThings/ThingType/v1/<thing type name>/Things(<thing ID>)

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <thngdtl.r>

### Request Header Parameters

Header	Required	Description
sap-iot-issync	No	Indicates whether the service retrieves the latest data creation time for the specified thing ID.  Permissible values are <b>true</b> or <b>false</b>

### Request Parameters

Parameter	Required	Data Type	Description
thing type name	Yes	String	Name of the thing type
thing ID	Yes	String	Unique identifier of a thing

### Request Example

```
/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/  
Things(0BCDD3EB71DB4935BAF9603577BF73F4)
```

### Response

#### Response Status and Error Codes

Code	Description
200	Details retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as depicted in the following examples.

### Example 1

```
/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/  
Things(0BCDD3EB71DB4935BAF9603577BF73F4)
```

Retrieves details of the thing 0BCDD3EB71DB4935BAF9603577BF73F4

```
{  
  "d": {  
    "__metadata": {  
      "id": "https://details-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/  
CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/  
Things('0BCDD3EB71DB4935BAF9603577BF73F4')",  
      "uri": "https://details-thing-sap-ci.cfapps.sap.hana.ondemand.com:  
443/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/  
Things('0BCDD3EB71DB4935BAF9603577BF73F4')",  
      "type": "com.sap.apptot.Thing"  
    },  
    "ThingId": "0BCDD3EB71DB4935BAF9603577BF73F4",  
    "ThingISOCODE": null,  
    "ThingDescription": "English",  
    "ThingName": "advlistThing1",  
    "ThingExternalId": null,  
    "ThingType": "test.sandbox.devxperf.package:BMW6Series",  
    "ObjectGroup": "FC132A31F78949C4809CE525841A06ED",  
    "BPName": null,  
    "BPLandlinePhoneNumber": "34343434",  
    "BPMobilePhoneNumber": null,  
    "ThingCustomerId": "671B285CCCCB42DE8241532745AD62E6",  
    "ThingRegionDescription": "Baden-Wuerttemberg",  
    "ThingCountryDescription": "Germany",  
    "ThingCountryId": "DE",  
    "ThingCommunicationData": "StreetName095 7, 69190 Walldorf",  
    "ThingLocationId": "4463FF2442C245369025582FE9AB0FDF",  
    "AlternateId": "advlistThing1_altId",  
    "DYN_ENT_core_automobiles__Model": {  
      "__deferred": {  
        "uri": "https://details-thing-sap-  
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/  
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/  
DYN_ENT_core_automobiles__Model"  
      }  
    },  
    "DYN_ENT_core_automobiles__Wheel": {  
      "__deferred": {  
        "uri": "https://details-thing-sap-  
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/  
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/  
DYN_ENT_core_automobiles__Wheel"  
      }  
    },  
    "DYN_ENT_core_automobiles__Engine": {  
      "__deferred": {  
        "uri": "https://details-thing-sap-  
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/  
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/  
DYN_ENT_core_automobiles__Engine"  
      }  
    }  
  }  
}
```

```

    },
    "DYN_ENT_core_automobiles__DriveUnit": {
      "__deferred": {
        "uri": "https://details-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/
DYN_ENT_core_automobiles__DriveUnit"
      }
    },
    "DYN_ENT_core_automobiles__LocationMaster": {
      "__deferred": {
        "uri": "https://details-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/
DYN_ENT_core_automobiles__LocationMaster"
      }
    }
  }
}

```

## Example 2

/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/  
Things(0BCDD3EB71DB4935BAF9603577BF73F4)

Request header parameter `sap-iot-issync` set to **true** and hence, the result set contains the last sync date of the thing.

Retrieves details of the thing 0BCDD3EB71DB4935BAF9603577BF73F4

```

{
  "d": {
    "__metadata": {
      "id": "https://details-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/
CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/
Things('0BCDD3EB71DB4935BAF9603577BF73F4')",
      "uri": "https://details-thing-sap-ci.cfapps.sap.hana.ondemand.com:
443/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/
Things('0BCDD3EB71DB4935BAF9603577BF73F4')",
      "type": "com.sap.apptot.Thing"
    },
    "ThingId": "0BCDD3EB71DB4935BAF9603577BF73F4",
    "ThingISOCCode": null,
    "ThingDescription": "English",
    "ThingName": "advlistThing1",
    "ThingExternalId": null,
    "ThingType": "test.sandbox.devxperf.package:BMW6Series",
    "ObjectGroup": "FC132A31F78949C4809CE525841A06ED",
    "BPName": null,
    "BPLandlinePhoneNumber": "34343434",
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": "671B285CCCCB42DE8241532745AD62E6",
    "ThingRegionDescription": "Baden-Wuerttemberg",
    "ThingCountryDescription": "Germany",
    "ThingCountryId": "DE",
    "ThingCommunicationData": "StreetName095 7, 69190 Walldorf",
    "ThingLocationId": "4463FF2442C245369025582FE9AB0FDF",
    "AlternateId": "advlistThing1_altId",
    "SyncedTimestamp": "2018-11-13T09:57:33.921Z",
    "DYN_ENT_core_automobiles__Model": {
      "__deferred": {
        "uri": "https://details-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/
DYN_ENT_core_automobiles__Model"
      }
    }
  }
}

```

```

    },
    "DYN_ENT_core_automobiles__Wheel": {
      "__deferred": {
        "uri": "https://details-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/
DYN_ENT_core_automobiles__Wheel"
      }
    },
    "DYN_ENT_core_automobiles__Engine": {
      "__deferred": {
        "uri": "https://details-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/
DYN_ENT_core_automobiles__Engine"
      }
    },
    "DYN_ENT_core_automobiles__DriveUnit": {
      "__deferred": {
        "uri": "https://details-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/
DYN_ENT_core_automobiles__DriveUnit"
      }
    },
    "DYN_ENT_core_automobiles__LocationMaster": {
      "__deferred": {
        "uri": "https://details-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/
DYN_ENT_core_automobiles__LocationMaster"
      }
    }
  }
}

```

## Related Information

[Thing Detail \[page 1121\]](#)

[Read Metadata of Thing Detail \[page 1122\]](#)

## 21.5.3 Read Property Set Data for a Thing

Retrieves property set data for the specified thing ID

### Request

**URI:** /CompositeThings/ThingType/v1/<thing type name>/Things(<thing ID>)

**Operation Type:** CRUD

**HTTP Method:** [GET](#)



**Permissions:** <thngdtl.r>

## Request Header Parameters

Header	Required	Description
sap-iot-issync	No	Indicates whether the service retrieves the latest data creation time for the specified thing ID.  Permissible values are <b>true</b> or <b>false</b>

## Request Parameters

Parameter	Required	Data Type	Description
thing type name	Yes	String	Name of the thing type
thing ID	Yes	String	Unique identifier of the thing

## Request Example

```
/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/  
Things(0BCDD3EB71DB4935BAF9603577BF73F4)?  
$expand=DYN_ENT_core_automobiles__Wheel,DYN_ENT_core_automobiles__Model
```

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

**Format:** *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure in the following examples.

### Example 1

```
/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/  
Things(0BCDD3EB71DB4935BAF9603577BF73F4)?  
$expand=DYN_ENT_core_automobiles__Wheel,DYN_ENT_core_automobiles__Model
```

```
{  
  "d": {  
    "__metadata": {
```

```

        "id": "https://details-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')",
        "uri": "https://details-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')",
        "type": "com.sap.apptot.Thing"
    },
    "ThingId": "0BCDD3EB71DB4935BAF9603577BF73F4",
    "ThingISOCode": null,
    "ThingDescription": "English",
    "ThingName": "thing_AWS_OBJ0045_095",
    "ThingExternalId": null,
    "ThingType": "core.automobiles:ABCXSeries",
    "ObjectGroup": "FC132A31F78949C4809CE525841A06ED",
    "BPName": null,
    "BPLandlinePhoneNumber": "34343434",
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": "671B285CCCCB42DE8241532745AD62E6",
    "ThingRegionDescription": "Baden-Wuerttemberg",
    "ThingCountryDescription": "Germany",
    "ThingCountryId": "DE",
    "ThingCommunicationData": "StreetName095 7, 69190 Walldorf",
    "ThingLocationId": "4463FF2442C245369025582FE9AB0FDF",
    "AlternateId": "thing_AWS_OBJ0045_095_altId",
    "DYN_ENT_core_automobiles_Model": {
        "ThingId": "0BCDD3EB71DB4935BAF9603577BF73F4",
        "modelData.Color": "Grey",
        "modelData.Color.Timestamp": "/Date(1489449600000)/",
        "modelData.Make": "Make2",
        "modelData.Make.Timestamp": "/Date(1489449600000)/",
        "modelData.Registration": "MYN 625",
        "modelData.Registration.Timestamp": "/Date(1489449600000)/"
    },
    "DYN_ENT_core_automobiles_Wheel": {
        "ThingId": "0BCDD3EB71DB4935BAF9603577BF73F4",
        "flWheel.PSI": 338,
        "flWheel.PSI.PressureUnitOfMeasure": null,
        "flWheel.PSI.PressureLowerThreshold": null,
        "flWheel.PSI.Timestamp": "/Date(1489622399000)/",
        "flWheel.PSI.PressureLowerThreshold.Timestamp": null,
        "flWheel.PSI.PressureUnitOfMeasure.Timestamp": null,
        "flWheel.Rotation": 234,
        "flWheel.Rotation.RotationUnitOfMeasure": null,
        "flWheel.Rotation.RotationLowerThreshold": null,
        "flWheel.Rotation.Timestamp": "/Date(1489622399000)/",
        "flWheel.Rotation.RotationUnitOfMeasure.Timestamp": null,
        "flWheel.Rotation.RotationLowerThreshold.Timestamp": null,
        "frWheel.PSI": 338,
        "frWheel.PSI.PressureUnitOfMeasure": null,
        "frWheel.PSI.PressureLowerThreshold": null,
        "frWheel.PSI.Timestamp": "/Date(1489622399000)/",
        "frWheel.PSI.PressureLowerThreshold.Timestamp": null,
        "frWheel.PSI.PressureUnitOfMeasure.Timestamp": null,
        "frWheel.Rotation": 234,
        "frWheel.Rotation.RotationUnitOfMeasure": null,
        "frWheel.Rotation.RotationLowerThreshold": null,
        "frWheel.Rotation.Timestamp": "/Date(1489622399000)/",
        "frWheel.Rotation.RotationUnitOfMeasure.Timestamp": null,
        "frWheel.Rotation.RotationLowerThreshold.Timestamp": null,
        "rlWheel.PSI": 338,
        "rlWheel.PSI.PressureUnitOfMeasure": null,
        "rlWheel.PSI.PressureLowerThreshold": null,
        "rlWheel.PSI.Timestamp": "/Date(1489622399000)/",
        "rlWheel.PSI.PressureUnitOfMeasure.Timestamp": null,
        "rlWheel.PSI.PressureLowerThreshold.Timestamp": null,
        "rlWheel.Rotation": 234,
        "rlWheel.Rotation.RotationLowerThreshold": null,
    }
}

```

```

        "rlWheel.Rotation.RotationUnitOfMeasure": null,
        "rlWheel.Rotation.Timestamp": "/Date(1489622399000)/",
        "rlWheel.Rotation.RotationUnitOfMeasure.Timestamp": null,
        "rlWheel.Rotation.RotationLowerThreshold.Timestamp": null,
        "rrWheel.PSI": 338,
        "rrWheel.PSI.PressureUnitOfMeasure": null,
        "rrWheel.PSI.PressureLowerThreshold": null,
        "rrWheel.PSI.Timestamp": "/Date(1489622399000)/",
        "rrWheel.PSI.PressureUnitOfMeasure.Timestamp": null,
        "rrWheel.PSI.PressureLowerThreshold.Timestamp": null,
        "rrWheel.Rotation": 234,
        "rrWheel.Rotation.RotationLowerThreshold": null,
        "rrWheel.Rotation.RotationUnitOfMeasure": null,
        "rrWheel.Rotation.Timestamp": "/Date(1489622399000)/",
        "rrWheel.Rotation.RotationUnitOfMeasure.Timestamp": null,
        "rrWheel.Rotation.RotationLowerThreshold.Timestamp": null
    },
    "DYN_ENT_core_automobiles__Engine": {
        "__deferred": {
            "uri": "https://details-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/
DYN_ENT_core_automobiles__Engine"
        }
    },
    "DYN_ENT_core_automobiles__DriveUnit": {
        "__deferred": {
            "uri": "https://details-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/
DYN_ENT_core_automobiles__DriveUnit"
        }
    },
    "DYN_ENT_core_automobiles__LocationMaster": {
        "__deferred": {
            "uri": "https://details-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/
core.automobiles:ABCXSeries/Things('0BCDD3EB71DB4935BAF9603577BF73F4')/
DYN_ENT_core_automobiles__LocationMaster"
        }
    }
}
}
}

```

### Example 2

```
/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/  
Things(C5D30DAAB52C4F01B14C169199BFE0CF)?$expand=DYN_ENT_core_automobiles_Wheel
```

Request header parameter `sap-iot-issync` is set **true** and hence, the result set contains the last sync date of the thing.

```
{
  "d": {
    "__metadata": {
      "id": "https://details-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/Things('C5D30DAAB52C4F01B14C169199BFE0CF')",
      "uri": "https://details-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/ThingType/v1/core.automobiles:ABCXSeries/Things('C5D30DAAB52C4F01B14C169199BFE0CF')",
      "type": "com.sap.apptot.Thing"
    },
    "ThingId": "C5D30DAAB52C4F01B14C169199BFE0CF",
    "ThingISOCODE": null,
    "ThingDescription": "vehicle thing description",
  }
}
```

```

    "ThingName": "aggreTextId1",
    "ThingExternalId": "aggreTextId1",
    "ThingType": "core.automobiles:ABCXSeries",
    "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
    "BPName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingCommunicationData": null,
    "ThingLocationId": null,
    "AlternateId": "aggreTextId1-altId",
    "SyncedTimestamp": "2018-11-13T09:57:33.921Z",
    "DYN_ENT_core_automobiles_Wheel": {
      "ThingId": "C5D30DAAB52C4F01B14C169199BFE0CF",
      "aggrepropset.pressure": 28,
      "aggrepropset.pressure.Timestamp": "/Date(1541475746000)/",
      "aggrepropset.speed": 108,
      "aggrepropset.speed.Timestamp": "/Date(1541475746000)/"
    }
  }
}

```

## Related Information

[Thing Detail \[page 1121\]](#)

[Read Metadata of Thing Detail \[page 1122\]](#)

[Read Details of a Thing \[page 1129\]](#)

## 21.6 Advanced Thing List

### Overview

The service retrieves list of things along with their location and customer details, based on thing types, property set types, and the event type specified in the request header. The service retrieves metadata and data for property set types common across specified thing types along with the aggregate of events for the specified event type.

#### → Recommendation

For performance reasons, we recommend to request not more than two property set types of each data category – *MasterData* and *TimeSeriesData* – while using this service.

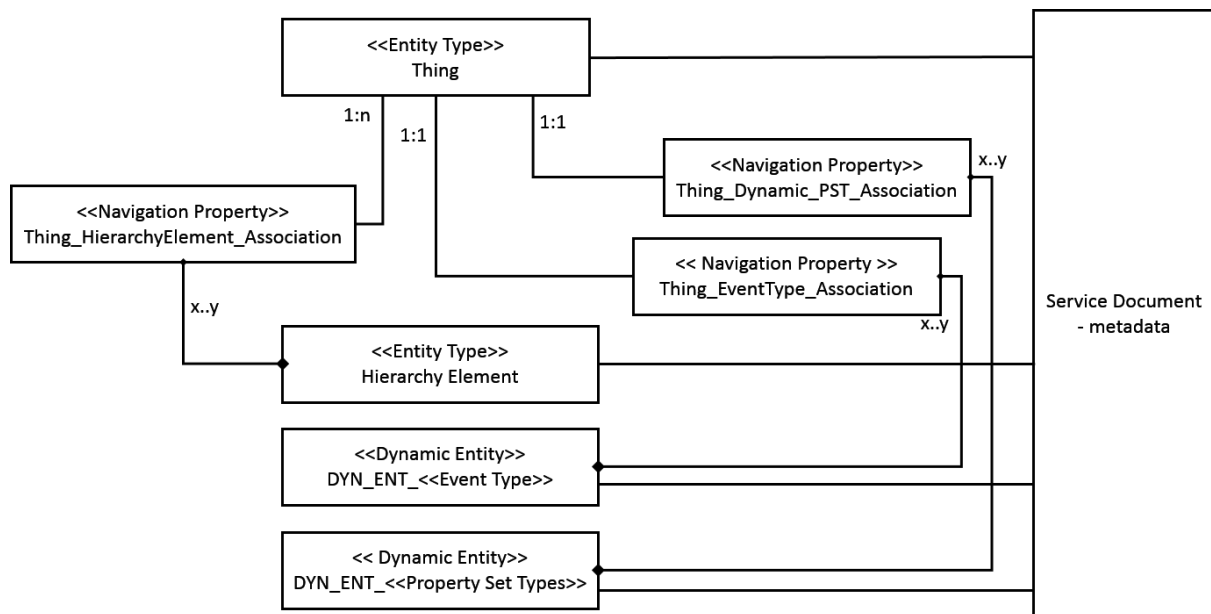
**OData Version: 2.0**

**Root URI:**http://<server address>[:<port number>]/CompositeThings/v1

**Example for a base URI in a cloud foundry environment:** https://advancedlist-thing-sap.cfapps.eu10.hana.ondemand.com/CompositeThings

## Entity Data Model

The following diagram shows the EDM of Thing List service:



## Related Information

[Read Metadata of Thing List \[page 1137\]](#)

[Read List of Things \[page 1148\]](#)

[Read Property Set Data and Events for a Thing \[page 1162\]](#)

### 21.6.1 Read Metadata of Thing List

Retrieves metadata of the thing list service

#### i Note

The service retrieves metadata for property set types common across specified thing types along with the aggregate of events for the specified event type.

- If you do not specify header parameters, the service retrieves only thing entity.

- If you specify both thing types and property set types in the header, the service retrieves metadata based only on the specified property set types.

The result set contains an annotation for the dynamic property set type entity. The annotation `sap:semantics="timeseries"` indicates that the property set type belongs to the data category [TimeSeriesData](#). Any property set type without an annotation indicates that it belongs to the data category [MasterData](#). In addition, the result set contains hierarchy details.

## Request

**URI:** `/CompositeThings/v1/$metadata`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:**

`<advlist>.r`

## Request Headers

Header	Required	Values
<code>sap-iot-thingtype</code>	No	Thing types separated by comma ",".
<code>sap-iot-pst</code>	No	Property set types separated by comma ",".
<code>sap-iot-eventtype</code>	No	Name of the event type
<div> <i>i</i> <b>Note</b>  Only one event type is supported. </div>		
<code>sap-iot-issync</code>	No	Indicates whether the service retrieves the latest data creation time for the specified thing ID.  Permissible values are <b>true</b> or <b>false</b>

## Request Parameters

None

## Request Example

`/CompositeThings/v1/$metadata`

## Response

### Response Status and Error Codes

Code	Description
200	Metadata retrieved successfully

## Payload

Only media type supported is [XML](#). The [XML](#) for this method has the structure as shown in the following examples:

**Example 1:** `/CompositeThings/v1/$metadata`

Request header parameter `sap-iot-issync` with value **true**

The result set contains the property `SyncedTimestamp` along with other relevant metadata.

```
<EntityType Name="HierarchyElement">
  <Key>
    <PropertyRef Name="HierarchyId"></PropertyRef>
    <PropertyRef Name="RootThingId"></PropertyRef>
  </Key>
  <Property Name="HierarchyId" Type="Edm.String"></Property>
  <Property Name="HierarchyName" Type="Edm.String"></Property>
  <Property Name="RootThingId" Type="Edm.String"></Property>
  <Property Name="RootThingName" Type="Edm.String"></Property>
  <Property Name="RootThingType" Type="Edm.String"></Property>
  <Property Name="RootThingDescription" Type="Edm.String"></
Property>
  <Property Name="RootThingExternalId" Type="Edm.String"></
Property>
  <Property Name="HierarchyDescription" Type="Edm.String"></
Property>
  <Property Name="HierarchyExternalId" Type="Edm.String"></
Property>
</EntityType>
<EntityType Name="Thing">
  <Key>
    <PropertyRef Name="ThingId"></PropertyRef>
  </Key>
  <Property Name="ThingId" Type="Edm.String" Nullable="false">
    <Documentation>
      <LongDescription>Thing unique ID</LongDescription>
    </Documentation>
  </Property>
  <Property Name="ISOCCode" Type="Edm.String">
    <Documentation>
      <LongDescription>Thing Language code</LongDescription>
    </Documentation>
  </Property>
  <Property Name="ThingDescription" Type="Edm.String">
    <Documentation>
      <LongDescription>Thing description</LongDescription>
    </Documentation>
  </Property>
  <Property Name="ThingName" Type="Edm.String" Nullable="false">
    <Documentation>
      <LongDescription>Thing name</LongDescription>
    </Documentation>
  </Property>
</EntityType>
```

```

        </Documentation>
    </Property>
    <Property Name="ThingExternalId" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing external id</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingType" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing Type ID associated with the
Thing</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ObjectGroup" Type="Edm.String">
        <Documentation>
            <LongDescription>Object group used to create the Thing</
LongDescription>
        </Documentation>
    </Property>
    <Property Name="BusinessPartnerName" Type="Edm.String">
        <Documentation>
            <LongDescription>Business partner name associated for
the Thing</LongDescription>
        </Documentation>
    </Property>
    <Property Name="BPLandlinePhoneNumber" Type="Edm.String">
        <Documentation>
            <LongDescription>Landline phone number of Business
partner</LongDescription>
        </Documentation>
    </Property>
    <Property Name="BPMobilePhoneNumber" Type="Edm.String">
        <Documentation>
            <LongDescription>Mobile phone number of Business
partner</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCustomerId" Type="Edm.String">
        <Documentation>
            <LongDescription>Business partner ID associated for the
Thing</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingRegionDescription" Type="Edm.String">
        <Documentation>
            <LongDescription>Region where Thing is located</
LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCountryDescription" Type="Edm.String">
        <Documentation>
            <LongDescription>Country where the Thing is located</
LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCountryId" Type="Edm.String">
        <Documentation>
            <LongDescription>Country code where the Thing is
located</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCommunicationData" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing address details</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingLocationId" Type="Edm.String">
        <Documentation>

```



```

        <LongDescription>Thing location ID</LongDescription>
    </Documentation>
</Property>
<Property Name="Latitude" Type="Edm.Double">
    <Documentation>
        <LongDescription>Latitude coordinates</LongDescription>
    </Documentation>
</Property>
<Property Name="Longitude" Type="Edm.Double">
    <Documentation>
        <LongDescription>Longitude coordinates</LongDescription>
    </Documentation>
</Property>
<Property Name="AlternateId" Type="Edm.String">
    <Documentation>
        <LongDescription>Thing AlternateId</LongDescription>
    </Documentation>
</Property>
<Property Name="SyncedTimestamp" Type="Edm.String">
    <Documentation>
        <LongDescription>Synced TimeStamp</LongDescription>
    </Documentation>
</Property>
    <NavigationProperty Name="HierarchyElements"
Relationship="com.sap.apptot.HierarchyElements" FromRole="ThingEntities"
ToRole="HierarchyElementsEntities"></NavigationProperty>
</EntityType>
    <Association Name="HierarchyElements">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.HierarchyElement" Multiplicity="*"
Role="HierarchyElementsEntities"></End>
    </Association>
    <EntityContainer Name="CompositeThingsAnalytics"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="HierarchyElements"
EntityType="com.sap.apptot.HierarchyElement"></EntitySet>
        <EntitySet Name="Things" EntityType="com.sap.apptot.Thing"></
EntitySet>
        <AssociationSet Name="HierarchyElements"
Association="com.sap.apptot.HierarchyElements">
            <End EntitySet="Things" Role="ThingEntities"></End>
            <End EntitySet="HierarchyElements"
Role="HierarchyElementsEntities"></End>
        </AssociationSet>
    </EntityContainer>

```

#### Example 2: /CompositeThings/v1/\$metadata

Request header parameter sap-iot-thingtype with value core.automobiles:ABCXSeries

The result set contains the property set type details associated with the specified thing type.

```

<EntityType Name="HierarchyElement">
    <Key>
        <PropertyRef Name="HierarchyId"></PropertyRef>
        <PropertyRef Name="RootThingId"></PropertyRef>
    </Key>
    <Property Name="HierarchyId" Type="Edm.String"></Property>
    <Property Name="HierarchyName" Type="Edm.String"></Property>
    <Property Name="RootThingId" Type="Edm.String"></Property>
    <Property Name="RootThingName" Type="Edm.String"></Property>
    <Property Name="RootThingType" Type="Edm.String"></Property>
    <Property Name="RootThingDescription" Type="Edm.String"></
Property>
    <Property Name="RootThingExternalId" Type="Edm.String"></
Property>

```

```

Property>      <Property Name="HierarchyDescription" Type="Edm.String"></
Property>      <Property Name="HierarchyExternalId" Type="Edm.String"></
Property>      </EntityType>
                <EntityType Name="Thing">
                    <Key>
                        <PropertyRef Name="ThingId"></PropertyRef>
                    </Key>
                    <Property Name="ThingId" Type="Edm.String" Nullable="false">
                        <Documentation>
                            <LongDescription>Thing unique ID</LongDescription>
                        </Documentation>
                    </Property>
                    <Property Name="ISOCODE" Type="Edm.String">
                        <Documentation>
                            <LongDescription>Thing Language code</LongDescription>
                        </Documentation>
                    </Property>
                    <Property Name="ThingDescription" Type="Edm.String">
                        <Documentation>
                            <LongDescription>Thing description</LongDescription>
                        </Documentation>
                    </Property>
                    <Property Name="ThingName" Type="Edm.String" Nullable="false">
                        <Documentation>
                            <LongDescription>Thing name</LongDescription>
                        </Documentation>
                    </Property>
                    <Property Name="ThingExternalId" Type="Edm.String">
                        <Documentation>
                            <LongDescription>Thing external id</LongDescription>
                        </Documentation>
                    </Property>
                    <Property Name="ThingType" Type="Edm.String">
                        <Documentation>
                            <LongDescription>Thing Type ID associated with the
Thing</LongDescription>
                        </Documentation>
                    </Property>
                    <Property Name="ObjectGroup" Type="Edm.String"> with value
                        <LongDescription>Object group used to create the Thing</
LongDescription>
                    </Documentation>
                    </Property>
                    <Property Name="BusinessPartnerName" Type="Edm.String">
                        <Documentation>
                            <LongDescription>Business partner name associated for
the Thing</LongDescription>
                        </Documentation>
                    </Property>
                    <Property Name="BPLandlinePhoneNumber" Type="Edm.String">
                        <Documentation>
                            <LongDescription>Landline phone number of Business
partner</LongDescription>
                        </Documentation>
                    </Property>
                    <Property Name="BPMobilePhoneNumber" Type="Edm.String">
                        <Documentation>
                            <LongDescription>Mobile phone number of Business
partner</LongDescription>
                        </Documentation>
                    </Property>
                    <Property Name="ThingCustomerId" Type="Edm.String">
                        <Documentation>
                            <LongDescription>Business partner ID associated for the
Thing</LongDescription>
                        </Documentation>

```

```

        </Property>
        <Property Name="ThingRegionDescription" Type="Edm.String">
            <Documentation>
                <LongDescription>Region where Thing is located</
LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingCountryDescription" Type="Edm.String">
            <Documentation>
                <LongDescription>Country where the Thing is located</
LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingCountryId" Type="Edm.String">
            <Documentation>
                <LongDescription>Country code where the Thing is
located</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingCommunicationData" Type="Edm.String">
            <Documentation>
                <LongDescription>Thing address details</LongDescription>
            </Documentation>
        </Property>
        <Property Name="ThingLocationId" Type="Edm.String">
            <Documentation>
                <LongDescription>Thing location ID</LongDescription>
            </Documentation>
        </Property>
        <Property Name="Latitude" Type="Edm.Double">
            <Documentation>
                <LongDescription>Latitude coordinates</LongDescription>
            </Documentation>
        </Property>
        <Property Name="Longitude" Type="Edm.Double">
            <Documentation>
                <LongDescription>Longitude coordinates</LongDescription>
            </Documentation>
        </Property>
        <Property Name="AlternateId" Type="Edm.String">
            <Documentation>
                <LongDescription>Thing AlternateId</LongDescription>
            </Documentation>
        </Property>
        <NavigationProperty Name="HierarchyElements"
Relationship="com.sap.apptot.HierarchyElements" FromRole="ThingEntities"
ToRole="HierarchyElementsEntities"></NavigationProperty>
        <NavigationProperty Name="DYN_ENT_core_automobiles__CarStyle"
Relationship="com.sap.apptot.DYN_ENT_core_automobiles__CarStyle"
FromRole="ThingEntities" ToRole="DYN_ENT_core_automobiles__CarStyleEntities"></
NavigationProperty>
        <NavigationProperty Name="DYN_ENT_core_automobiles__CarInfo"
Relationship="com.sap.apptot.DYN_ENT_core_automobiles__CarInfo"
FromRole="ThingEntities" ToRole="DYN_ENT_core_automobiles__CarInfoEntities"></
NavigationProperty>
        <NavigationProperty Name="DYN_ENT_core_automobiles__Wheel"
Relationship="com.sap.apptot.DYN_ENT_core_automobiles__Wheel"
FromRole="ThingEntities" ToRole="DYN_ENT_core_automobiles__WheelEntities"></
NavigationProperty>
    </EntityType>
    <EntityType Name="DYN_ENT_core_automobiles__Wheel"
sap:semantics="timeseries">
        <Key>
            <PropertyRef Name="ThingId"></PropertyRef>
        </Key>
        <Property Name="ThingId" Type="Edm.String"></Property>
        <Property Name="wheeltemp.temperature" Type="Edm.Int32"></
Property>

```

```

        <Property Name="wheeltemp.temperature.Timestamp"
Type="Edm.DateTimeOffset"></Property>
    </EntityType>
    <EntityType Name="DYN_ENT_core_automobiles__CarStyle">
        <Key>
            <PropertyRef Name="ThingId"></PropertyRef>
        </Key>
        <Property Name="ThingId" Type="Edm.String"></Property>
        <Property Name="carModel.Model" Type="Edm.Int32"></Property>
        <Property Name="carModel.Model.Timestamp"
Type="Edm.DateTimeOffset"></Property>
        <Property Name="carModel.color" Type="Edm.String"></Property>
        <Property Name="carModel.color.Timestamp"
Type="Edm.DateTimeOffset"></Property>
    </EntityType>
    <EntityType Name="DYN_ENT_core_automobiles__CarInfo">
        <Key>
            <PropertyRef Name="ThingId"></PropertyRef>
        </Key>
        <Property Name="ThingId" Type="Edm.String"></Property>
        <Property Name="carYear.year" Type="Edm.DateTime"></Property>
        <Property Name="carYear.year.Timestamp"
Type="Edm.DateTimeOffset"></Property>
    </EntityType>
    <Association Name="HierarchyElements">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.HierarchyElement" Multiplicity="*"
Role="HierarchyElementsEntities"></End>
    </Association>
    <Association Name="DYN_ENT_core_automobiles__CarInfo">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.DYN_ENT_core_automobiles__CarInfo"
Multiplicity="1" Role="DYN_ENT_core_automobiles__CarInfoEntities"></End>
    </Association>
    <Association Name="DYN_ENT_core_automobiles__CarStyle">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.DYN_ENT_core_automobiles__CarStyle"
Multiplicity="1" Role="DYN_ENT_core_automobiles__CarStyleEntities"></End>
    </Association>
    <Association Name="DYN_ENT_core_automobiles__Wheel">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.DYN_ENT_core_automobiles__Wheel"
Multiplicity="1" Role="DYN_ENT_core_automobiles__WheelEntities"></End>
    </Association>
    <EntityContainer Name="CompositeThingsAnalytics"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="HierarchyElements"
EntityType="com.sap.apptot.HierarchyElement"></EntitySet>
        <EntitySet Name="Things" EntityType="com.sap.apptot.Thing"></
EntitySet>
        <EntitySet Name="DYN_ENT_core_automobiles__Wheels"
EntityType="com.sap.apptot.DYN_ENT_core_automobiles__Wheel"></EntitySet>
        <EntitySet Name="DYN_ENT_core_automobiles__CarInfos"
EntityType="com.sap.apptot.DYN_ENT_core_automobiles__CarInfo"></EntitySet>
        <EntitySet Name="DYN_ENT_core_automobiles__CarStyles"
EntityType="com.sap.apptot.DYN_ENT_core_automobiles__CarStyle"></EntitySet>
        <AssociationSet Name="HierarchyElements"
Association="com.sap.apptot.HierarchyElements">
            <End EntitySet="Things" Role="ThingEntities"></End>
            <End EntitySet="HierarchyElements"
Role="HierarchyElementsEntities"></End>
        </AssociationSet>
        <AssociationSet Name="DYN_ENT_core_automobiles__CarInfo"
Association="com.sap.apptot.DYN_ENT_core_automobiles__CarInfo">

```

```

        <End EntitySet="Things" Role="ThingEntities"></End>
        <End EntitySet="DYN_ENT_core_automobiles__CarInfos"
Role="DYN_ENT_core_automobiles__CarInfoEntities"></End>
    </AssociationSet>
    <AssociationSet Name="DYN_ENT_core_automobiles__CarStyle"
Association="com.sap.apptot.DYN_ENT_core_automobiles__CarStyle">
        <End EntitySet="Things" Role="ThingEntities"></End>
        <End EntitySet="DYN_ENT_core_automobiles__CarStyles"
Role="DYN_ENT_core_automobiles__CarStyleEntities"></End>
    </AssociationSet>
    <AssociationSet Name="DYN_ENT_core_automobiles__Wheel"
Association="com.sap.apptot.DYN_ENT_core_automobiles__Wheel">
        <End EntitySet="Things" Role="ThingEntities"></End>
        <End EntitySet="DYN_ENT_core_automobiles__Wheels"
Role="DYN_ENT_core_automobiles__WheelEntities"></End>
    </AssociationSet>
</EntityContainer>

```

**Example 3:** /CompositeThings/v1/\$metadata

Request header parameters sap-iot-thingtypecore.automobiles:ABCXSeries and sap-iot-pst with value core.automobiles:Wheel

The result set contains the property set type details specified in the request header. The ting type name specified in the request header is not considered.

```

Property Name="ObjectGroup" Type="Edm.String" <EntityType
Name="HierarchyElement">
    <Key>
        <PropertyRef Name="HierarchyId"></PropertyRef>
        <PropertyRef Name="RootThingId"></PropertyRef>
    </Key>
    <Property Name="HierarchyId" Type="Edm.String"></Property>
    <Property Name="HierarchyName" Type="Edm.String"></Property>
    <Property Name="RootThingId" Type="Edm.String"></Property>
    <Property Name="RootThingName" Type="Edm.String"></Property>
    <Property Name="RootThingType" Type="Edm.String"></Property>
    <Property Name="RootThingDescription" Type="Edm.String"></
Property>
    <Property Name="RootThingExternalId" Type="Edm.String"></
Property>
    <Property Name="HierarchyDescription" Type="Edm.String"></
Property>
    <Property Name="HierarchyExternalId" Type="Edm.String"></
Property>
</EntityType>
<EntityType Name="Thing">
    <Key>
        <PropertyRef Name="ThingId"></PropertyRef>
    </Key>
    <Property Name="ThingId" Type="Edm.String" Nullable="false">
        <Documentation>
            <LongDescription>Thing unique ID</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ISOCODE" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing Language code</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingDescription" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing description</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingName" Type="Edm.String" Nullable="false">

```

```

        <Documentation>
            <LongDescription>Thing name</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingExternalId" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing external id</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingType" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing Type ID associated with the
Thing</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ObjectGroup" Type="Edm.String">
        <Documentation>
            <LongDescription>Object group used to create the Thing</
LongDescription>
        </Documentation>
    </Property>
    <Property Name="BusinessPartnerName" Type="Edm.String">
        <Documentation>
            <LongDescription>Business partner name associated for
the Thing</LongDescription>
        </Documentation>
    </Property>
    <Property Name="BPLandlinePhoneNumber" Type="Edm.String">
        <Documentation>
            <LongDescription>Landline phone number of Business
partner</LongDescription>
        </Documentation>
    </Property>
    <Property Name="BPMobilePhoneNumber" Type="Edm.String">
        <Documentation>
            <LongDescription>Mobile phone number of Business
partner</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCustomerId" Type="Edm.String">
        <Documentation>
            <LongDescription>Business partner ID associated for the
Thing</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingRegionDescription" Type="Edm.String">
        <Documentation>
            <LongDescription>Region where Thing is located</
LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCountryDescription" Type="Edm.String">
        <Documentation>
            <LongDescription>Country where the Thing is located</
LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCountryId" Type="Edm.String">
        <Documentation>
            <LongDescription>Country code where the Thing is
located</LongDescription>
        </Documentation>
    </Property>
    <Property Name="ThingCommunicationData" Type="Edm.String">
        <Documentation>
            <LongDescription>Thing address details</LongDescription>
        </Documentation>
    </Property>

```

```

        <Property Name="ThingLocationId" Type="Edm.String">
            <Documentation>
                <LongDescription>Thing location ID</LongDescription>
            </Documentation>
        </Property>
        <Property Name="Latitude" Type="Edm.Double">
            <Documentation>
                <LongDescription>Latitude coordinates</LongDescription>
            </Documentation>
        </Property>
        <Property Name="Longitude" Type="Edm.Double">
            <Documentation>
                <LongDescription>Longitude coordinates</LongDescription>
            </Documentation>
        </Property>
        <Property Name="AlternateId" Type="Edm.String">
            <Documentation>
                <LongDescription>Thing AlternateId</LongDescription>
            </Documentation>
        </Property>
        <NavigationProperty Name="HierarchyElements"
Relationship="com.sap.apptot.HierarchyElements" FromRole="ThingEntities"
ToRole="HierarchyElementsEntities"></NavigationProperty>
        <NavigationProperty Name="DYN_ENT_core_automobiles__Wheel"
Relationship="com.sap.apptot.DYN_ENT_core_automobiles__Wheel"
FromRole="ThingEntities" ToRole="DYN_ENT_core_automobiles__WheelEntities"></
NavigationProperty>
    </EntityType>
    <EntityType Name="DYN_ENT_core_automobiles__Wheel"
sap:semantics="timeseries">
        <Key>
            <PropertyRef Name="ThingId"></PropertyRef>
        </Key>
        <Property Name="ThingId" Type="Edm.String"></Property>
        <Property Name="wheeltemp.temperature" Type="Edm.Int32"></
Property>
        <Property Name="wheeltemp.temperature.Timestamp"
Type="Edm.DateTimeOffset"></Property>
    </EntityType>
    <Association Name="HierarchyElements">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.HierarchyElement" Multiplicity="*"
Role="HierarchyElementsEntities"></End>
    </Association>
    <Association Name="DYN_ENT_core_automobiles__Wheel">
        <End Type="com.sap.apptot.Thing" Multiplicity="1"
Role="ThingEntities"></End>
        <End Type="com.sap.apptot.DYN_ENT_core_automobiles__Wheel"
Multiplicity="1" Role="DYN_ENT_core_automobiles__WheelEntities"></End>
    </Association>
    <EntityContainer Name="CompositeThingsAnalytics"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="HierarchyElements"
EntityType="com.sap.apptot.HierarchyElement"></EntitySet>
        <EntitySet Name="Things" EntityType="com.sap.apptot.Thing"></
EntitySet>
        <EntitySet Name="DYN_ENT_core_automobiles__Wheels"
EntityType="com.sap.apptot.DYN_ENT_core_automobiles__Wheel"></EntitySet>
        <AssociationSet Name="HierarchyElements"
Association="com.sap.apptot.HierarchyElements">
            <End EntitySet="Things" Role="ThingEntities"></End>
            <End EntitySet="HierarchyElements"
Role="HierarchyElementsEntities"></End>
        </AssociationSet>
        <AssociationSet Name="DYN_ENT_core_automobiles__Wheel"
Association="com.sap.apptot.DYN_ENT_core_automobiles__Wheel">
            <End EntitySet="Things" Role="ThingEntities"></End>

```

```

        <End EntitySet="DYN_ENT_core_automobiles__Wheels"
Role="DYN_ENT_core_automobiles__WheelEntities"></End>
    </AssociationSet>
</EntityContainer>

```

## 21.6.2 Read List of Things

Retrieves list of things for the specified property set types and thing types.

With this service, you can retrieve list of things based on the following request header parameters:

- Thing type – You can list one or more thing types separated by comma.  
Service retrieves all things that belong to the listed thing types.
- Property set type – You can list one or more property set types separated by comma.  
Service retrieves all things of thing type that uses the specified property set types.
- Event type – You can specify only one event type.  
Service retrieves all things for which events are created for the specified event type.

By default, the service displays a maximum of 100 records per response. You can navigate to the next set of results, if available, using the `__next` link at the end in the response payload. However, the number of records retrieved depends on the query parameter `$top`, if provided in the request URL. For more information, see [Retrieving List of Records \[page 1220\]](#).

### Request

**URI:** `/CompositeThings/v1/Things`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:**

`<advlist>.r`

### Request Headers

Header	Required	Values
<code>sap-iot-thingtype</code>	No	Thing types separated by comma ",".
<code>sap-iot-pst</code>	No	Property set types separated by comma ",".
<code>sap-iot-eventtype</code>	No	Name of the event Type

**Note**

Only one event type is supported.



Header	Required	Values
sap-iot-issync	No	Indicates whether the service retrieves the latest data creation time for the specified thing ID.  Permissible values are <b>true</b> or <b>false</b>

The request header parameters are optional. You can specify values for one or more, or none of these parameters. In addition, the service also supports `$expand` query parameter in the request URL with which you can refine your result set. However, when the values provided in the request header parameter and in the `$expand` query parameter differ, the values specified in the `$expand` query parameter takes the precedence and result set is retrieved only based on them. The request header parameters are ignored in such cases.

The following table illustrates the available things, thing types, and property set types:

Things	Thing Type	Property Set Type
T1, T2	TT1	PST1
		PST2
T3, T4	TT2	PST2
		PST3
		PST4
T5	TT3	PST1
		PST2
		PST4

The following table illustrates how data is retrieved for a given combination of request header parameters and `$expand` query parameters:

Request	Request Header Parameters	Result set contains...	Based on the...
/things	TT1, TT2	T1, T2, T3, T4	Request header parameters
/things? \$expand=PST1	TT1	T1, T2, T5	\$expand query parameters (request header parameters are ignored)
/things? \$expand=PST2	TT1, TT2	T1, T2, T3, T4, T5	\$expand query parameters (request header parameters are ignored)
/things	PST1, PST2	T1, T2	Request header parameters
/things? \$expand=PST2	PST1, PST2	T1, T2, T3, T4, T5	\$expand query parameters (request header parameters are ignored)
/things? \$expand=PST1, PST2	PST1, PST2	T1, T2	Request header parameters and \$expand query parameters are same

## Request Parameters

None

### Query Request Parameters

Parameter	Required	Type	Description
<code>\$top</code>	No	Integer	Number of records to include in the result set.
<code>\$skip</code>	No	Integer	Number of the first n records to be excluded from the result set.
<code>\$inlinecount</code>	No	String	<p>Total number of things available; permissible values are <a href="#">allpages</a> and <a href="#">none</a>. The following results apply if this parameter is used in conjunction with the other query parameters:</p> <ul style="list-style-type: none"><li>• If you use this with the <code>\$top</code> and <code>\$skip</code> parameters, the result contains the total count of things available.</li><li>• If you use this with the <code>\$filter</code> parameter, the result contains the total count of things based on the defined filter condition.</li></ul>
<code>\$count</code>	No	Boolean	<p>Total number of things available based on the current user's authorizations; permissible values are true and false.</p> <p>If this is used in conjunction with the <code>\$top</code> and <code>\$skip</code> parameters, the result contains the total count of things available.</p>

Parameter	Required	Type	Description
\$orderby	No	String	Field name to be used for sorting the result set in ascending order; all thing entity fields are valid.  To sort the result set, use <i>asc</i> for the ascending order and <i>desc</i> for the descending order.
\$select		String	Select condition for the thing entity fields to be included in the result set.
\$filter		String	Filter condition to be applied; all thing entity fields are valid.  Multiple filters are supported using the <i>and</i> separator.  You can use the string functions such as <i>substringof</i> , <i>startswith</i> , and <i>endswith</i> to search data using \$filter.
sap-apptot-es			Predefined query parameter used to filter events based on the event status. The only supported field is status.  As the status field is not included in the metadata, you can use this parameter to filter events based on status.  For example, sap-apptot-es=Status eq 'inprocess'

## Request Example

/CompositeThings/v1/Things

Assume, there are 1000 records in the database. Due to performance reasons, the method retrieves the first 100 records and displays a next link to navigate to the next set of records.

/CompositeThings/v1/Things?\$top=500

Assume, there are 1000 records in the database. Due to performance reasons, the method retrieves the first 100 records and displays a next link to navigate to the next set of records until top 500 records are retrieved.

/CompositeThings/v1/Things?\$select=Latitude, Longitude, ThingId

Assume, there are 1000 records in the database. Due to performance reasons, the method retrieves the first 100 records and displays a next link to navigate to the next set of records. The response payload contains only the fields specified in the `$select` query parameter.

```
/CompositeThings/v1/Things?$select=Latitude, Longitude, ThingId&$top=100
```

Assume, there are 1000 records in the database. The method retrieves top 100 records and displays only those fields specified in the `$select` query parameter.

## Response

### Response Status and Error Codes

Code	Reason
200	Thing list retrieved successfully.

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the structure as depicted in the following examples.

### Example 1

Request header parameter `sap-iot-thingtype` with value `core.automobiles:ABCXSeries`

The result set contains all things that belong to the thing type specified in the request header.

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://advancedlist-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')",
          "uri": "https://advancedlist-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')",
          "type": "com.sap.apptot.Thing"
        },
        "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
        "ISOCODE": "en",
        "ThingDescription": "advanced list thing 1 description",
        "ThingName": "advlistThing1",
        "ThingExternalId": "advlistThing1",
        "ThingType": "core.automobiles:ABCXSeries",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
      }
    ]
  }
}
```

```

        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing1-altId",
        "HierarchyElements": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/HierarchyElements"
            }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/DYN_ENT_core_automobiles__CarStyle"
            }
        },
        "DYN_ENT_core_automobiles__Wheel": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/DYN_ENT_core_automobiles__Wheel"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BBA340F5CC264347ACFCFD6B66F049EA')",
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BBA340F5CC264347ACFCFD6B66F049EA')",
            "type": "com.sap.apptot.Thing"
        },
        "ThingId": "BBA340F5CC264347ACFCFD6B66F049EA",
        "ISOCODE": "en",
        "ThingDescription": "advanced list thing 12 description",
        "ThingName": "advlistThing3",
        "ThingExternalId": "advlistThing12",
        "ThingType": "core.automobiles:ABCXSeries",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing12-altId",
        "HierarchyElements": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BBA340F5CC264347ACFCFD6B66F049EA')/HierarchyElements"
            }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
            "__deferred": {

```

```

        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BBA340F5CC264347ACFCFD6B66F049EA')/DYN_ENT_core_automobiles__CarStyle"
    },
    "DYN_ENT_core_automobiles__Wheel": {
        "__deferred": {
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BBA340F5CC264347ACFCFD6B66F049EA')/DYN_ENT_core_automobiles__Wheel"
        }
    }
}
]
}
}

```

## Example 2

Request header parameters sap-iot-thingtype with values core.automobiles:ABCXSeries and core.automobiles:ABCXSeries2

The result set contains all things that belong to the thing types specified in the request header.

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
          "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
          "type": "com.sap.apptot.Thing"
        },
        "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
        "ISOCCode": "en",
        "ThingDescription": "advanced list thing 1 description",
        "ThingName": "advlistThing1",
        "ThingExternalId": "advlistThing1",
        "ThingType": "core.automobiles:ABCXSeries",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing1-altId",
        "HierarchyElements": {
          "__deferred": {
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/HierarchyElements"
          }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
          "__deferred": {

```

```

        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/DYN_ENT_core_automobiles__CarStyle"
    }
    },
    {
        "__metadata": {
            "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')",
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')",
            "type": "com.sap.apptot.Thing"
        },
        "ThingId": "B354E59948B6486ABA70734ED15F793A",
        "ISOCODE": "en",
        "ThingDescription": "advanced list thing 1 description",
        "ThingName": "advlistThing2",
        "ThingExternalId": "advlistThing2",
        "ThingType": "core.automobiles:ABCXSeries2",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing2-altId",
        "HierarchyElements": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')/HierarchyElements"
            }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')/DYN_ENT_core_automobiles__CarStyle"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BBA340F5CC264347ACFCFD6B66F049EA')",
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BBA340F5CC264347ACFCFD6B66F049EA')",
            "type": "com.sap.apptot.Thing"
        },
        "ThingId": "BBA340F5CC264347ACFCFD6B66F049EA",
        "ISOCODE": "en",
        "ThingDescription": "advanced list thing 12 description",
        "ThingName": "advlistThing3",
        "ThingExternalId": "advlistThing12",
        "ThingType": "core.automobiles:ABCXSeries",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,

```

```

        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing12-altId",
        "HierarchyElements": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BBA340F5CC264347ACFCFD6B66F049EA')/HierarchyElements"
            }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BBA340F5CC264347ACFCFD6B66F049EA')/DYN_ENT_core_automobiles__CarStyle"
            }
        }
    },
    {
        "__metadata": {
            "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')",
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')",
            "type": "com.sap.apptot.Thing"
        },
        "ThingId": "BE2639D6E7EE4C9D8073C30B5A33BAA1",
        "ISOCODE": "en",
        "ThingDescription": "advanced list thing 22 description",
        "ThingName": "advlistThing3",
        "ThingExternalId": "advlistThing22",
        "ThingType": "core.automobiles:ABCXSeries",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing22-altId",
        "HierarchyElements": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')/HierarchyElements"
            }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')/DYN_ENT_core_automobiles__CarStyle"
            }
        }
    }
}

```



```

    }
  ]
}

```

### Example 3

Request header parameters `sap-iot-thingtype` with value `core.automobiles:ABCXSeries2`

Request URL with `$expand` query parameter: `/CompositeThings/v1/Things?`

`$expand=DYN_ENT_core_automobiles__Wheel`

The result set contains things that belong to thing types using the property set type specified in the `$expand` query parameter in the request URL. The thing type specified in the request header is ignored.

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('9922491EB8A94A77B2FF1ACAE00C28C4')",
          "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('9922491EB8A94A77B2FF1ACAE00C28C4')",
          "type": "com.sap.apptot.Thing"
        },
        "ThingId": "9922491EB8A94A77B2FF1ACAE00C28C4",
        "ISOCODE": "en",
        "ThingDescription": "advanced list thing 3 description",
        "ThingName": "advlistThing3",
        "ThingExternalId": "advlistThing3",
        "ThingType": "core.automobiles:ABCXSeries3",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing3-altId",
        "HierarchyElements": {
          "__deferred": {
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('9922491EB8A94A77B2FF1ACAE00C28C4')/HierarchyElements"
          }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
          "__deferred": {
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('9922491EB8A94A77B2FF1ACAE00C28C4')/DYN_ENT_core_automobiles__CarStyle"
          }
        },
        "DYN_ENT_core_automobiles__CarInfo": {
          "__deferred": {

```

```

        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('9922491EB8A94A77B2FF1ACAE00C28C4')/DYN_ENT_core_automobiles__CarInfo"
    },
    "DYN_ENT_core_automobiles__Wheel": {
        "ThingId": "9922491EB8A94A77B2FF1ACAE00C28C4",
        "wheeltemp.temperature": null,
        "wheeltemp.temperature.Timestamp": null
    },
    {
        "__metadata": {
            "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')",
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')",
            "type": "com.sap.apiot.Thing"
        },
        "ThingId": "B354E59948B6486ABA70734ED15F793A",
        "ISOCODE": "en",
        "ThingDescription": "advanced list thing 1 description",
        "ThingName": "advlistThing2",
        "ThingExternalId": "advlistThing2",
        "ThingType": "core.automobiles:ABCXSeries32",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing2-altId",
        "HierarchyElements": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')/HierarchyElements"
            }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')/DYN_ENT_core_automobiles__CarStyle"
            }
        },
        "DYN_ENT_core_automobiles__CarInfo": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')/DYN_ENT_core_automobiles__CarInfo"
            }
        },
        "DYN_ENT_core_automobiles__Wheel": {
            "ThingId": "B354E59948B6486ABA70734ED15F793A",
            "wheeltemp.temperature": null,
            "wheeltemp.temperature.Timestamp": null
        }
    },
    {

```

```

        "__metadata": {
            "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')",
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')",
            "type": "com.sap.apptot.Thing"
        },
        "ThingId": "BE2639D6E7EE4C9D8073C30B5A33BAA1",
        "ISOCODE": "en",
        "ThingDescription": "advanced list thing 22 description",
        "ThingName": "advlistThing3",
        "ThingExternalId": "advlistThing22",
        "ThingType": "iotae.citest1.core.advlistpkg:advlistTT2",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing22-altId",
        "HierarchyElements": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')/HierarchyElements"
            }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')/DYN_ENT_core_automobiles__CarStyle"
            }
        },
        "DYN_ENT_core_automobiles__CarInfo": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')/DYN_ENT_core_automobiles__CarInfo"
            }
        },
        "DYN_ENT_core_automobiles__Wheel": {
            "ThingId": "BE2639D6E7EE4C9D8073C30B5A33BAA1",
            "wheeltemp.temperature": null,
            "wheeltemp.temperature.Timestamp": null
        }
    }
}
]
}
}

```

#### Example 4

Request header parameters `sap-iot-thingtype` with value `core.automobiles:ABCXSeries2` and `sap-iot-issync` with value **true**

Request URL with `$expand` query parameter: `/CompositeThings/v1/Things?`  
`$expand=DYN_ENT_core_automobiles__Wheel`

The result set contains things that belong to thing types using the property set type specified in the \$expand query parameter in the request URL. The thing type specified in the request header is ignored. The last sync time is included in the result set as the request parameter sap-iot-issync is set **true**.

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://advancedlist-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/Things('9922491EB8A94A77B2FF1ACAE00C28C4')",
          "uri": "https://advancedlist-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/Things('9922491EB8A94A77B2FF1ACAE00C28C4')",
          "type": "com.sap.appiot.Thing"
        },
        "ThingId": "9922491EB8A94A77B2FF1ACAE00C28C4",
        "ISOCODE": "en",
        "ThingDescription": "advanced list thing 3 description",
        "ThingName": "advlistThing3",
        "ThingExternalId": "advlistThing3",
        "ThingType": "core.automobiles:ABCXSeries3",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing3-altId",
        "SyncedTimestamp": "2018-11-13T09:57:33.921Z",
        "HierarchyElements": {
          "__deferred": {
            "uri": "https://advancedlist-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/Things('9922491EB8A94A77B2FF1ACAE00C28C4')/HierarchyElements"
          }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
          "__deferred": {
            "uri": "https://advancedlist-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/Things('9922491EB8A94A77B2FF1ACAE00C28C4')/DYN_ENT_core_automobiles__CarStyle"
          }
        },
        "DYN_ENT_core_automobiles__CarInfo": {
          "__deferred": {
            "uri": "https://advancedlist-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/Things('9922491EB8A94A77B2FF1ACAE00C28C4')/DYN_ENT_core_automobiles__CarInfo"
          }
        },
        "DYN_ENT_core_automobiles__Wheel": {
          "ThingId": "9922491EB8A94A77B2FF1ACAE00C28C4",
          "wheeltemp.temperature": 66,
          "wheeltemp.temperature.Timestamp": "/Date(1541475746000)/"
        }
      },
      {
        "__metadata": {
```

```

        "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')",
        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')",
        "type": "com.sap.apptot.Thing"
    },
    "ThingId": "B354E59948B6486ABA70734ED15F793A",
    "ISOCODE": "en",
    "ThingDescription": "advanced list thing 1 description",
    "ThingName": "advlistThing2",
    "ThingExternalId": "advlistThing2",
    "ThingType": "core.automobiles:ABCXSeries32",
    "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
    "BusinessPartnerName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingCommunicationData": null,
    "ThingLocationId": null,
    "Latitude": null,
    "Longitude": null,
    "AlternateId": "advlistThing2-altId",
    "SyncedTimestamp": null,
    "HierarchyElements": {
        "__deferred": {
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')/HierarchyElements"
        }
    },
    "DYN_ENT_core_automobiles__CarStyle": {
        "__deferred": {
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')/DYN_ENT_core_automobiles__CarStyle"
        }
    },
    "DYN_ENT_core_automobiles__CarInfo": {
        "__deferred": {
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('B354E59948B6486ABA70734ED15F793A')/DYN_ENT_core_automobiles__CarInfo"
        }
    },
    "DYN_ENT_core_automobiles__Wheel": {
        "ThingId": "B354E59948B6486ABA70734ED15F793A",
        "wheeltemp.temperature": 45,
        "wheeltemp.temperature.Timestamp": "/Date(1541475746000)/"
    }
},
{
    "__metadata": {
        "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')",
        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')",
        "type": "com.sap.apptot.Thing"
    },
    "ThingId": "BE2639D6E7EE4C9D8073C30B5A33BAA1",
    "ISOCODE": "en",
    "ThingDescription": "advanced list thing 22 description",

```

```

        "ThingName": "advlistThing3",
        "ThingExternalId": "advlistThing22",
        "ThingType": "iotae.citest1.core.advlistpkg:advlistTT2",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing22-altId",
        "SyncedTimestamp": null,
        "HierarchyElements": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')/HierarchyElements"
            }
        },
        "DYN_ENT_core_automobiles__CarStyle": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')/DYN_ENT_core_automobiles__CarStyle"
            }
        },
        "DYN_ENT_core_automobiles__CarInfo": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('BE2639D6E7EE4C9D8073C30B5A33BAA1')/DYN_ENT_core_automobiles__CarInfo"
            }
        },
        "DYN_ENT_core_automobiles__Wheel": {
            "ThingId": "BE2639D6E7EE4C9D8073C30B5A33BAA1",
            "wheeltemp.temperature": 60,
            "wheeltemp.temperature.Timestamp": "/Date(1541475746000)/"
        }
    }
}
]
}
}

```

### 21.6.3 Read Property Set Data and Events for a Thing

Retrieves property set data and aggregate of events for the specified thing

You can use this service to retrieve the details of the specified thing that has a usage of the specified property set type. The result set includes details of the specified thing along with its property set data and aggregate of events. You can also retrieve the details of the hierarchy to which the thing belongs.

## Request

**URI:** /CompositeThings/v1/Things('<thing ID>')

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <advlist>.r

### Request Headers

Header	Required	Values
sap-iot-thingtype	No	Thing types separated by comma ","
sap-iot-pst	No	Property set types separated by comma ","
sap-iot-eventtype	No	Name of the event type
<div><div>i Note</div><div>Only one event type is supported.</div></div>		
sap-iot-issync	No	Indicates whether the service retrieves the latest data creation time for the specified thing ID.  Permissible values are <b>true</b> or <b>false</b>

For more information on the various combination usage of the header parameters, see the **Request Headers** section in the topic [Read List of Things \[page 1148\]](#).

### Request Parameters

Parameter	Required	Data Type	Description
thing ID	Yes	String	Unique identifier of a thing

### Request Example

```
/CompositeThings/v1/Things('00045BA79B50488781EBBA3BC1C8D384')
```

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

**Format:** *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the structure as depicted in the following examples::

### Example 1

```
/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')
```

Retrieves composite thing details of the specified thing ID.

```
{
  "d": {
    "__metadata": {
      "id": "https://advancedlist-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')",
      "uri": "https://advancedlist-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')",
      "type": "com.sap.apptot.Thing"
    },
    "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
    "ISOCCode": "en",
    "ThingDescription": "advanced list thing 1 description",
    "ThingName": "advlistThing1",
    "ThingExternalId": "advlistThing1",
    "ThingType": "core.automobiles:ABCXSeries",
    "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
    "BusinessPartnerName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingCommunicationData": null,
    "ThingLocationId": null,
    "Latitude": null,
    "Longitude": null,
    "AlternateId": "advlistThing1-altId",
    "HierarchyElements": {
      "__deferred": {
        "uri": "https://advancedlist-thing-sap-ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')/HierarchyElements"
      }
    }
  }
}
```

### Example 2

```
/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')
```

Request header parameter `sap-iot-eventtypewith` with value `core.automobiles:advlistET1`

Retrieves composite thing details for the specified thing ID along with the details of the dynamic attributes expanded.

```
{
  "d": {
```



```

    "__metadata": {
      "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "type": "com.sap.apptot.Thing"
    },
    "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
    "ISOCODE": "en",
    "ThingDescription": "advanced list thing 1 description",
    "ThingName": "advlistThing1",
    "ThingExternalId": "advlistThing1",
    "ThingType": "core.automobiles:ABCXSeries",
    "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
    "BusinessPartnerName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingCommunicationData": null,
    "ThingLocationId": null,
    "Latitude": null,
    "Longitude": null,
    "AlternateId": "advlistThing1-altId",
    "HierarchyElements": {
      "__deferred": {
        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/HierarchyElements"
      }
    },
    "DYN_ENT_core_automobiles__advlistET1": {
      "__deferred": {
        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/DYN_ENT__core_automobiles__advlistET1"
      }
    }
  }
}

```

### Example 3

/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')?  
\$expand=DYN\_ENT\_core\_automobiles\_\_advlistET1

Request header parameter sap-iot-eventtype with value  
iotae.citest1.core.advlistpkg:advlistET1

The result set contains the aggregate of events corresponding to the specified thing ID.

```

{
  "d": {
    "__metadata": {
      "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "type": "com.sap.apptot.Thing"
    },

```

```

    "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
    "ISOCCode": "en",
    "ThingDescription": "advanced list thing 1 description",
    "ThingName": "advlistThing1",
    "ThingExternalId": "advlistThing1",
    "ThingType": "core.automobiles:ABCXSeries",
    "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
    "BusinessPartnerName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingCommunicationData": null,
    "ThingLocationId": null,
    "Latitude": null,
    "Longitude": null,
    "AlternateId": "advlistThing1-altId",
    "HierarchyElements": {
      "__deferred": {
        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/HierarchyElements"
      }
    },
    "DYN_ENT_core_automobiles_advlistET1": {
      "EventType": "core.automobiles:advlistET1",
      "High": 1,
      "Medium": 1,
      "Low": null
    }
  }
}

```

#### Example 4

```

/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')?
$expand=DYN_ENT_core_automobiles__advlistET1

```

Request header parameters sap-iot-eventtype with value core.automobiles:advlistET1 and sap-iot-thingtype with value core.automobiles:ABCXSeries

```

{
  "d": {
    "__metadata": {
      "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "type": "com.sap.apptot.Thing"
    },
    "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
    "ISOCCode": "en",
    "ThingDescription": "advanced list thing 1 description",
    "ThingName": "advlistThing1",
    "ThingExternalId": "advlistThing1",
    "ThingType": "core.automobiles:ABCXSeries",
    "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
    "BusinessPartnerName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
  }
}

```

```

    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingCommunicationData": null,
    "ThingLocationId": null,
    "Latitude": null,
    "Longitude": null,
    "AlternateId": "advlistThing1-altId",
    "HierarchyElements": {
      "__deferred": {
        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/HierarchyElements"
      }
    },
    "DYN_ENT_core_automobiles__CarStyle": {
      "__deferred": {
        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/DYN_ENT_core_automobiles__CarStyle"
      }
    },
    "DYN_ENT_core_automobiles__Engine": {
      "__deferred": {
        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/DYN_ENT_core_automobiles__Engine"
      }
    },
    "DYN_ENT_core_automobiles__advlistET1": {
      "EventType": "core.automobiles:advlistET1",
      "High": 1,
      "Medium": 1,
      "Low": null
    }
  }
}

```

### Example 5

```

/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')?
$expand=DYN_ENT_core_automobiles__advlistET1, DYN_ENT_core_automobiles__Engine,
DYN_ENT_core_automobiles__CarStyle

```

Request header parameters sap-iot-eventtype with value core.automobiles:advlistET1 and sap-iot-thingtype with value core.automobiles:ABCXSeries

```

{
  "d": {
    "__metadata": {
      "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "type": "com.sap.apptot.Thing"
    },
    "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
    "ISOCCode": "en",
    "ThingDescription": "advanced list thing 1 description",
    "ThingName": "advlistThing1",
    "ThingExternalId": "advlistThing1",
    "ThingType": "core.automobiles:ABCXSeries",
    "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
    "BusinessPartnerName": null,
    "BPLandlinePhoneNumber": null,

```

```

    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingCommunicationData": null,
    "ThingLocationId": null,
    "Latitude": null,
    "Longitude": null,
    "AlternateId": "advlistThing1-altId",
    "HierarchyElements": {
      "__deferred": {
        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/HierarchyElements"
      }
    },
    "DYN_ENT_core_automobiles_CarStyle": {
      "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
      "CarModel.Model": null,
      "CarModel.Model.Timestamp": null,
      "CarModel.color": null,
      "CarModel.color.Timestamp": null
    },
    "DYN_ENT_core_automobiles_Engine": {
      "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
      "SpepedPressure.pressure": 28,
      "SpepedPressure.pressure.Timestamp": "/Date(1527465600000)/",
      "SpepedPressure.speed": 108,
      "SpepedPressure.speed.Timestamp": "/Date(1527465600000)/"
    },
    "DYN_ENT_core_automobiles__advlistET1": {
      "EventType": "core.automobiles:advlistET1",
      "High": 1,
      "Medium": 1,
      "Low": null
    }
  }
}

```

## Example 6

```

/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')?
$expand=HierarchyElements

```

Retrieves composite thing details along with details of the hierarchy to which things belong. The following response is in [JSON](#) format.

```

{
  "d": {
    "__metadata": {
      "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "type": "com.sap.apiot.Thing"
    },
    "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
    "ISOCODE": "en",
    "ThingDescription": "advanced list thing 1 description",
    "ThingName": "advlistThing1",
    "ThingExternalId": "advlistThing1",
    "ThingType": "core.automobiles:ABCXSeries",
    "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
  }
}

```

```

    "BusinessPartnerName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
    "ThingRegionDescription": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingCommunicationData": null,
    "ThingLocationId": null,
    "Latitude": null,
    "Longitude": null,
    "AlternateId": "advlistThing1-altId",
    "HierarchyElements": {
      "results": [
        {
          "HierarchyId": "A23E0110DA6CA5471600B1023FD0F1AC",
          "HierarchyName": "advlist_Hierarchy",
          "RootThingId": "B354E59948B6486ABA70734ED15F793A",
          "RootThingName": "advlistThing2",
          "RootThingType": "core.automobiles:ABCX2Series",
          "RootThingDescription": "advanced list thing 2 description",
          "RootThingExternalId": "advlistThing2",
          "HierarchyDescription": "Desccribes hierarchy of units in
the carXYZ",
          "HierarchyExternalId": "advlist-Hierarchy"
        },
        {
          "HierarchyId": "B13E0110DA6CA5471600B1023FD0F1AC",
          "HierarchyName": "advlist_Hierarchy",
          "RootThingId": "B354E59948B6486ABA70734ED15F793A",
          "RootThingName": "advlistThing2",
          "RootThingType": "core.automobiles:ABCX2Series",
          "RootThingDescription": "advanced list thing 2 description",
          "RootThingExternalId": "advlistThing2",
          "HierarchyDescription": "Desccribes hierarchy of units in
the carXYZ",
          "HierarchyExternalId": "advlist-Hierarchy"
        }
      ]
    }
  }
}

```

### Example 7

/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')/HierarchyElements

Retrieves hierarchy elements for the specified thing ID.

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
HierarchyElements(HierarchyId='A23E0110DA6CA5471600B1023FD0F1AC',RootThingId='B35
4E59948B6486ABA70734ED15F793A')",
          "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
HierarchyElements(HierarchyId='A23E0110DA6CA5471600B1023FD0F1AC',RootThingId='B35
4E59948B6486ABA70734ED15F793A')",
          "type": "com.sap.apptot.HierarchyElement"
        },
        "HierarchyId": "A23E0110DA6CA5471600B1023FD0F1AC",
        "HierarchyName": "advlist_Hierarchy",
        "RootThingId": "B354E59948B6486ABA70734ED15F793A",

```

```

        "RootThingName": "advlistThing2",
        "RootThingType": "core.automobiles:ABCX2Series",
        "RootThingDescription": "advanced list thing 2 description",
        "RootThingExternalId": "advlistThing2",
        "HierarchyDescription": "Desccribes hierarchy of units in the
carXYZ",
        "HierarchyExternalId": "advlist-Hierarchy"
    },
    {
        "__metadata": {
            "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
HierarchyElements(HierarchyId='B13E0110DA6CA5471600B1023FD0F1AC',RootThingId='B35
4E59948B6486ABA70734ED15F793A')",
            "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
HierarchyElements(HierarchyId='B13E0110DA6CA5471600B1023FD0F1AC',RootThingId='B35
4E59948B6486ABA70734ED15F793A')",
            "type": "com.sap.apptiot.HierarchyElement"
        },
        "HierarchyId": "B13E0110DA6CA5471600B1023FD0F1AC",
        "HierarchyName": "advlist_Hierarchy",
        "RootThingId": "B354E59948B6486ABA70734ED15F793A",
        "RootThingName": "advlistThing2",
        "RootThingType": "core.automobiles:ABCX2Series",
        "RootThingDescription": "advanced list thing 2 description",
        "RootThingExternalId": "advlistThing2",
        "HierarchyDescription": "Desccribes hierarchy of units in the
carXYZ",
        "HierarchyExternalId": "advlist-Hierarchy"
    }
]
}
}

```

## Example 8

```

/CompositeThings/v1/Things('0E121F2106A94BCBB1D2413D4575386A')?
$expand=DYN_ENT_core_automobiles__advlistET1,DYN_ENT_core_automobiles__Engine,DYN_E
NT_core_automobiles__CarStyle

```

Request header parameters `sap-iot-eventtype` with value `core.automobiles:advlistET1`, `sap-iot-thingtype` with value `core.automobiles:ABCXSeries`, and `sap-iot-issync` with value **true**.

```

{
  "d": {
    "__metadata": {
      "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')",
      "type": "com.sap.apptiot.Thing"
    },
    "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
    "ISOCCode": "en",
    "ThingDescription": "advanced list thing 1 description",
    "ThingName": "advlistThing1",
    "ThingExternalId": "advlistThing1",
    "ThingType": "core.automobiles:ABCXSeries",
    "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
    "BusinessPartnerName": null,
    "BPLandlinePhoneNumber": null,
    "BPMobilePhoneNumber": null,
    "ThingCustomerId": null,
  }
}

```

```

    "ThingRegionDescription": null,
    "ThingCountryDescription": null,
    "ThingCountryId": null,
    "ThingCommunicationData": null,
    "ThingLocationId": null,
    "Latitude": null,
    "Longitude": null,
    "AlternateId": "advlistThing1-altId",
    "SyncedTimestamp": "2018-11-13T09:57:33.921Z",
    "HierarchyElements": {
      "__deferred": {
        "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('0E121F2106A94BCBB1D2413D4575386A')/HierarchyElements"
      }
    },
    "DYN_ENT_core_automobiles__CarStyle": {
      "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
      "CarModel.Model": null,
      "CarModel.Model.Timestamp": null,
      "CarModel.color": null,
      "CarModel.color.Timestamp": null
    },
    "DYN_ENT_core_automobiles__Engine": {
      "ThingId": "0E121F2106A94BCBB1D2413D4575386A",
      "SpepedPressure.pressure": 28,
      "SpepedPressure.pressure.Timestamp": "/Date(1527465600000)/",
      "SpepedPressure.speed": 108,
      "SpepedPressure.speed.Timestamp": "/Date(1527465600000)/"
    },
    "DYN_ENT_core_automobiles__advlistET1": {
      "EventType": "core.automobiles:advlistET1",
      "High": 1,
      "Medium": 1,
      "Low": null
    }
  }
}

```

### Example 9

You can filter things based on a dynamic property value as explained in this example. /

```

CompositeThings/v1/Things?$expand=DYN_ENT_core_automobiles__CarStyle&
$filter=DYN_ENT_core_automobiles__CarStyle/CarModel.Model eq '2019T'

```

Request header parameters `sap-iot-eventtype` with value `core.automobiles:advlistET1`, `sap-iot-thingtype` with value `core.automobiles:ABCXSeries`.

```

{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('F244DA0C48EE416699732D4F1F9C5008')",
          "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('F244DA0C48EE416699732D4F1F9C5008')",
          "type": "com.sap.apptot.Thing"
        },
        "ThingId": "F244DA0C48EE416699732D4F1F9C5008",
        "ISOCode": "en",
        "ThingDescription": "advanced list thing 2 description",
        "ThingName": "advlistThing2",
        "ThingExternalId": "advlistThing2",

```

```

        "ThingType": "core.automobiles:ABCXSeries",
        "ObjectGroup": "47D746ECC8A049CABB98E9D266044FD4",
        "BusinessPartnerName": null,
        "BPLandlinePhoneNumber": null,
        "BPMobilePhoneNumber": null,
        "ThingCustomerId": null,
        "ThingRegionDescription": null,
        "ThingCountryDescription": null,
        "ThingCountryId": null,
        "ThingCommunicationData": null,
        "ThingLocationId": null,
        "Latitude": null,
        "Longitude": null,
        "AlternateId": "advlistThing2-altId",
        "DYN_ENT_core_automobiles__CarStyle": {
            "ThingId": "F244DA0C48EE416699732D4F1F9C5008",
            "CarModel.model": "2019T",
            "CarModel.model.Timestamp": "/Date(1562144400000)/"
        },
        "HierarchyElements": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('F244DA0C48EE416699732D4F1F9C5008')/HierarchyElements"
            }
        },
        "DYN_ENT_core_automobiles__Engine": {
            "__deferred": {
                "uri": "https://advancedlist-thing-sap-
ci.cfapps.sap.hana.ondemand.com:443/CompositeThings/v1/
Things('F244DA0C48EE416699732D4F1F9C5008')/DYN_ENT_core_automobiles__Engine"
            }
        }
    }
}
]
}
}

```

## 21.7 Event List: OData Service

### Overview

The event list OData service is used to retrieve the complete information of events for a specific event type.

#### OData Version: 2.0

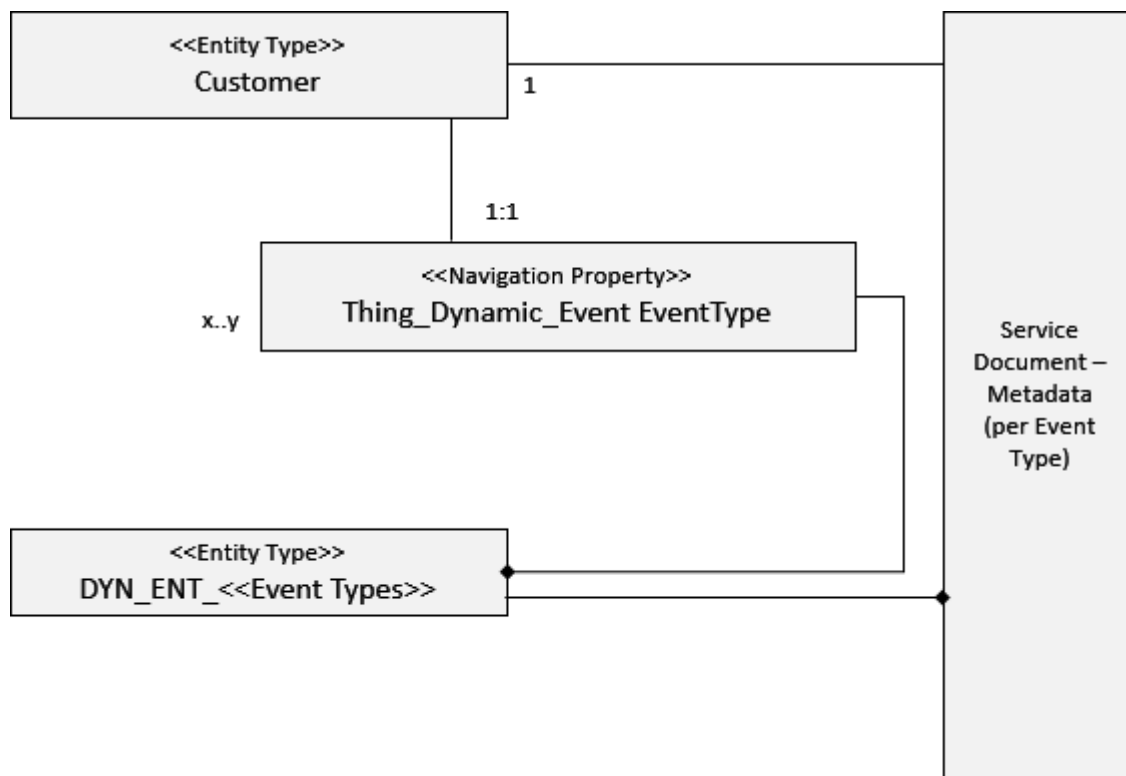
**Root URI:** `http://<server address>[:<port number>]/BPAnalytics/EventType/v1/<event type name>`

**Example for a Root URI in a cloud foundry environment:** `https://bpanalytics-event-sap.cfapps.eu10.hana.ondemand.com/BPAnalytics/EventType/v1/<event type name>`



## Entity Data Model

The following diagram shows the EDM of Event List OData service:



**Service Metadata URI:** `http://<server address>[:<port number>]/BPAnalytics/EventType/v1/<event type name>/$metadata`

## Related Information

[Read Metadata of Event Type \[page 1173\]](#)

[Read Events of Event Type for Customer \[page 1175\]](#)

### 21.7.1 Read Metadata of Event Type

Retrieves metadata of a specific event type

#### i Note

You cannot specify multiple event types in the request URL.

## Request

**URI:** /BPAnalytics/EventType/v1/<event type name>/\$metadata

**Operation Type:** CRUD

**HTTP Method:** *GET*

**Permissions:** <bpanlyt>.r

### Request Parameters

Parameter	Required	Data Type	Description
event type name	Yes	String	Name of the event type

### Request Example

/BPAnalytics/EventType/v1/core.automobiles:AlertEventType/\$metadata

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

```
<EntityType Name="Customer">
  <Key>
    <PropertyRef Name="BusinessPartnerId"/>
  </Key>
  <Property Name="BusinessPartnerId" Type="Edm.String"/>
  <Property Name="Name" Type="Edm.String"/>
  <Property Name="Country" Type="Edm.String" Nullable="false"/>
  <Property Name="AddressRepresentation" Type="Edm.String"/>
  <Property Name="LandlinePhoneNumber" Type="Edm.String"/>
  <Property Name="MobilePhoneNumber" Type="Edm.String"/>
  <NavigationProperty Name="DYN_ENT_core_automobiles__AlertEventType"
Relationship="com.sap.apptot.odata2.bpevents.DYN_ENT_core_automobiles__AlertEvent
Type" FromRole="CustomerEntities"
ToRole="DYN_ENT_core_automobiles_AlertEventTypeEntities">
</EntityType>
<EntityType Name="DYN_ENT_core_automobiles__AlertEventType">
  <Key>
    <PropertyRef Name="EventTypeId"/>
  </Key>
  <Property Name="EventTypeId" Type="Edm.String" Nullable="false"/>
  <Property Name="High" Type="Edm.Numeric"/>
  <Property Name="Medium" Type="Edm.Numeric"/>
  <Property Name="Low" Type="Edm.Numeric"/>
</EntityType>
```

```
<EntityContainer Name="CompositeThingStatisticDataContainer"
m:IsDefaultEntityContainer="true">
  <EntitySet Name="Customer"
EntityType="com.sap.appiot.odata2.bpevents.Customers"/>
  <EntitySet Name="DYN_ENT_core_automobiles__AlertEventType"
EntityType="com.sap.appiot.odata2.bpevents.DYN_ENT_core_automobiles__AlertEventTy
pe"/>
</EntityContainer>
```

## Related Information

[Event List: OData Service \[page 1172\]](#)

[Read Events of Event Type for Customer \[page 1175\]](#)

## 21.7.2 Read Events of Event Type for Customer

Retrieves all events of a specific event type

You can use this service to specify an event type and retrieve aggregates of events for each severity. The service filters the events based on the status and groups the events by customer and their location. The service retrieves all customers for which valid things are created in the system and the respective locations of the customers.

### Request

**URI:** /BPAnalytics/EventType/v1/<event type name>/Customers

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <bpanlyt>.r

### Request Headers

Header	Required	Description
Accept-Language	No	Language of the event type description. The default language is <a href="#">en</a> .

## Request Parameters

Parameter	Required	Data Type	Description
event type name	Yes	String	Name of the event type.
<div><b>i Note</b> You cannot specify multiple event types in the request URL.</div>			
customer ID	Yes	String	Unique identifier of the customer

## Query Request Parameters

Parameter	Required	Data Type	Description
\$top	No	Integer	Number of records to include in the result set.
<div><b>i Note</b> Default value is 20</div>			
\$skip	No	Integer	Number of the first n records to be excluded from the result set.
\$orderby	No	String	<p>Field name to be used for sorting the result set in ascending order; valid fields are business partner ID, name, country, and country description.</p> <p>To sort the result set, use <a href="#">asc</a> for the ascending order and <a href="#">desc</a> for the descending order.</p>
\$filter	No	String	<p>Filter condition to be applied; valid fields are business partner ID, name, country, and country description.</p> <p>Multiple filters are supported using the <a href="#">AND</a> separator.</p>
\$select	No	String	Select condition for the data fields to be included in the result set; valid fields are business partner ID, name, country, and country description.

Parameter	Required	Data Type	Description
sap-appiot-es	No	String	<p>Predefined query parameter used to filter events based on the event status. The only supported field is <code>Status</code>.</p> <p>As the status field is not included in the metadata, you can use this parameter to filter events based on <code>Status</code>. For example, <code>sap-appiot-es=Status eq 'coming'</code>.</p>

## Request Example

```
/BPAnalytics/EventType/V1/core.automobiles:AlertEventType/Customers
```

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [XML](#)

Media types supported are [JSON](#) and [XML](#). The [XML](#) for this method has the structure as depicted in the following examples:

```
/BPAnalytics/EventType/V1/core.automobiles:AlertEventType/Customers
```

Retrieves all events of a specific event type along with customer details and their respective location details

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://
schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://
schemas.microsoft.com/ado/2007/08/dataservices" xml:base="https://bpanalytics-
event-sap-ci.cfapps.sap.hana.ondemand.com:443/BPAnalytics/EventType/v1/
core.automobiles:AlertEventType/">
  <id>https://bpanalytics-event-sap-ci.cfapps.sap.hana.ondemand.com:443/
BPAnalytics/EventType/v1/core.automobiles:AlertEventType/Customers</id>
  <title type="text">Customers</title>
  <updated>2017-04-04T05:13:49.478Z</updated>
  <author>
    <name></name>
  </author>
  <link href="Customers" rel="self" title="Customers"></link>
```

```

<entry>
  <id>https://bpanalytics-event-sap-ci.cfapps.sap.hana.ondemand.com:443/
BPAnalytics/EventType/v1/core.automobiles:AlertEventType/
Customers('022ACCDE8BD242939AEB70FF42D9133C')</id>
  <title type="text">Customers</title>
  <updated>2017-04-04T05:13:49.486Z</updated>
  <category term="com.sap.apptot.Customers" scheme="http://
schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category>
  <link href="Customers('022ACCDE8BD242939AEB70FF42D9133C')" rel="edit"
title="Customers"></link>
  <link href="Customers('022ACCDE8BD242939AEB70FF42D9133C')/
DYN_ENT_core_automobiles__AlertEventType" rel="http://schemas.microsoft.com/ado/
2007/08/dataservices/related/DYN_ENT_core_automobiles__AlertEventType"
title="DYN_ENT_core_automobiles__AlertEventType" type="application/atom
+xml;type=entry"></link>
  <content type="application/xml">
    <m:properties>
      <d:BusinessPartnerId>022ACCDE8BD242939AEB70FF42D9133C</
d:BusinessPartnerId>
      <d:Name>Stephan</d:Name>
      <d:Country>FR</d:Country>
      <d:CountryDescription>France</d:CountryDescription>
      <d:BPAddressRepresentationISOCODE>Zyyy</
d:BPAddressRepresentationISOCODE>
      <d:AddressRepresentation>Ring 7, 99999 Walldorf</
d:AddressRepresentation>
      <d:LandlinePhoneNumber>34343434</d:LandlinePhoneNumber>
      <d:MobilePhoneNumber m:null="true"></d:MobilePhoneNumber>
    </m:properties>
  </content>
</entry>
<entry>
  <id>https://bpanalytics-event-sap-ci.cfapps.sap.hana.ondemand.com:443/
BPAnalytics/EventType/v1/core.automobiles:AlertEventType/
Customers('1D8A662167F0427AB02D837D975B12EA')</id>
  <title type="text">Customers</title>
  <updated>2017-04-04T05:13:49.494Z</updated>
  <category term="com.sap.apptot.Customers" scheme="http://
schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category>
  <link href="Customers('1D8A662167F0427AB02D837D975B12EA')" rel="edit"
title="Customers"></link>
  <link href="Customers('1D8A662167F0427AB02D837D975B12EA')/
DYN_ENT_core_automobiles__AlertEventType" rel="http://schemas.microsoft.com/ado/
2007/08/dataservices/related/DYN_ENT_core_automobiles__AlertEventType"
title="DYN_ENT_core_automobiles__AlertEventType" type="application/atom
+xml;type=entry"></link>
  <content type="application/xml">
    <m:properties>
      <d:BusinessPartnerId>1D8A662167F0427AB02D837D975B12EA</
d:BusinessPartnerId>
      <d:Name>WildCardSearch WildCard</d:Name>
      <d:Country>DE</d:Country>
      <d:CountryDescription>Germany</d:CountryDescription>
      <d:BPAddressRepresentationISOCODE>Zyyy</
d:BPAddressRepresentationISOCODE>
      <d:AddressRepresentation>7, 77777 Walldorf</
d:AddressRepresentation>
      <d:LandlinePhoneNumber>34343434</d:LandlinePhoneNumber>
      <d:MobilePhoneNumber m:null="true"></d:MobilePhoneNumber>
    </m:properties>
  </content>
</entry>
</feed>

```

```

/BPAnalytics/EventType/V1/core.automobiles:AlertEventType/
Customers('022ACCDE8BD242939AEB70FF42D9133C')

```

Retrieves all events of a specific event type along with customer details and their respective location details for the specified customer

```
<?xml version="1.0" encoding="utf-8"?>
<entry xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://
schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://
schemas.microsoft.com/ado/2007/08/dataservices" xml:base="https://bpanalytics-
event-sap-ci.cfapps.sap.hana.ondemand.com:443/BPAAnalytics/EventType/v1/
core.automobiles:AlertEventType/">
  <id>https://bpanalytics-event-sap-ci.cfapps.sap.hana.ondemand.com:443/
BPAAnalytics/EventType/v1/core.automobiles:AlertEventType/
Customers('022ACCDE8BD242939AEB70FF42D9133C')</id>
  <title type="text">Customers</title>
  <updated>2017-04-04T05:17:14.584Z</updated>
  <category term="com.sap.apptot.Customers" scheme="http://
schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category>
  <link href="Customers('022ACCDE8BD242939AEB70FF42D9133C')" rel="edit"
title="Customers"></link>
  <link href="Customers('022ACCDE8BD242939AEB70FF42D9133C') /
DYN_ENT_core_automobiles__AlertEventType" rel="http://schemas.microsoft.com/ado/
2007/08/dataservices/related/DYN_ENT_core_automobiles__AlertEventType"
title="DYN_ENT_core_automobiles__AlertEventType" type="application/atom
+xml;type=entry"></link>
  <content type="application/xml">
    <m:properties>
      <d:BusinessPartnerId>022ACCDE8BD242939AEB70FF42D9133C</
d:BusinessPartnerId>
      <d:Name>Stephan</d:Name>
      <d:Country>FR</d:Country>
      <d:CountryDescription>France</d:CountryDescription>
      <d:BPAAddressRepresentationISOCODE>Zyyy</
d:BPAAddressRepresentationISOCODE>
      <d:AddressRepresentation>Ring 7, 99999 Walldorf</
d:AddressRepresentation>
      <d:LandlinePhoneNumber>34343434</d:LandlinePhoneNumber>
      <d:MobilePhoneNumber m:null="true"></d:MobilePhoneNumber>
    </m:properties>
  </content>
</entry>
```

/BPAAnalytics/EventType/V1/core.automobiles:AlertEventType/Customers?  
\$expand=DYN\_ENT\_core\_automobiles\_\_AlertEventType

Retrieves all events of a specific event type along with customer details and their respective location details. In addition, the service lists aggregates of events for each severity.

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:m="http://
schemas.microsoft.com/ado/2007/08/dataservices/metadata" xmlns:d="http://
schemas.microsoft.com/ado/2007/08/dataservices" xml:base="https://bpanalytics-
event-sap-ci.cfapps.sap.hana.ondemand.com:443/BPAAnalytics/EventType/v1/
core.automobiles:AlertEventType/">
  <id>https://bpanalytics-event-sap-ci.cfapps.sap.hana.ondemand.com:443/
BPAAnalytics/EventType/v1/core.automobiles:AlertEventType/Customers</id>
  <title type="text">Customers</title>
  <updated>2017-04-04T05:19:33.919Z</updated>
  <author>
    <name></name>
  </author>
  <link href="Customers" rel="self" title="Customers"></link>
  <entry>
    <id>https://bpanalytics-event-sap-ci.cfapps.sap.hana.ondemand.com:443/
BPAAnalytics/EventType/v1/core.automobiles:AlertEventType/
Customers('022ACCDE8BD242939AEB70FF42D9133C')</id>
    <title type="text">Customers</title>
    <updated>2017-04-04T05:19:33.919Z</updated>
```

```

    <category term="com.sap.apptot.Customers" scheme="http://
schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category>
    <link href="Customers('022ACCDE8BD242939AEB70FF42D9133C')" rel="edit"
title="Customers"></link>
    <link href="Customers('022ACCDE8BD242939AEB70FF42D9133C')/
DYN_ENT_core_automobiles__AlertEventType" rel="http://schemas.microsoft.com/ado/
2007/08/dataservices/related/DYN_ENT_core_automobiles__AlertEventType"
title="DYN_ENT_core_automobiles__AlertEventType" type="application/atom
+xml;type=entry">
        <m:inline>
            <entry xml:base="https://bpanalytics-event-sap-
ci.cfapps.sap.hana.ondemand.com:443/BPAnalytics/EventType/v1/
core.automobiles:AlertEventType/">
                <id>https://bpanalytics-event-sap-
ci.cfapps.sap.hana.ondemand.com:443/BPAnalytics/EventType/v1/
core.automobiles:AlertEventType/
DYN_ENT_core_automobiles__AlertEventType('test.sandbox.devxperf.eventpackage
%3AAlertEventType')</id>
                <title type="text">DYN_ENT_core_automobiles__AlertEventType</
title>
                <updated>2017-04-04T05:19:33.921Z</updated>
                <category
term="com.sap.apptot.DYN_ENT_test_core_automobiles__AlertEventType"
scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category>
                <link
href="DYN_ENT_core_automobiles__AlertEventType('core.automobiles
%3AAlertEventType')" rel="edit"
title="DYN_ENT_core_automobiles__AlertEventType"></link>
                <content type="application/xml">
                    <m:properties>
                        <d:EventTypeId>core.automobiles:AlertEventType</
d:EventTypeId>
                        <d:High>10</d:High>
                        <d:Medium>2</d:Medium>
                        <d:Low>4</d:Low>
                    </m:properties>
                </content>
            </entry>
        </m:inline>
    </link>
    <content type="application/xml">
        <m:properties>
            <d:BusinessPartnerId>022ACCDE8BD242939AEB70FF42D9133C</
d:BusinessPartnerId>
            <d:Name>Stephan</d:Name>
            <d:Country>FR</d:Country>
            <d:CountryDescription>France</d:CountryDescription>
            <d:BPAddressRepresentationISOCODE>Zyyy</
d:BPAddressRepresentationISOCODE>
            <d:AddressRepresentation>Ring 7, 99999 Walldorf</
d:AddressRepresentation>
            <d:LandlinePhoneNumber>34343434</d:LandlinePhoneNumber>
            <d:MobilePhoneNumber m:null="true"></d:MobilePhoneNumber>
        </m:properties>
    </content>
</entry>
</feed>

```

## Related Information

[Event List: OData Service \[page 1172\]](#)

[Read Metadata of Event Type \[page 1173\]](#)



## 21.8 Events Aggregate: OData Service

### Overview

The events aggregate OData service is used to retrieve aggregate of events grouped by the event type for the specified time interval.

#### Time Bucket

The time bucket is the time interval that is calculated as follows:

Time Bucket = (toTime - fromTime) / number of divisions

#### OData Version: 2.0

**Root URI:** `http://<server address>[:<port number>]/EventsAggregate/v1/Events?timeRange=<from time>-<to time>&divisions=<divisions>`

**Example for a Root URI in a cloud foundry environment:** `https://events-aggregate-sap.cfapps.eu10.hana.ondemand.com/EventsAggregate/v1/Events?timeRange=<from time>-<to time>&divisions=<divisions>`

#### Support of OData Features

The current implementation supports only \$filter query option, which can be passed as query or path parameter.

**Service Metadata URI:** `http://<server address>[:<port number>]/EventsAggregate/v1/$metadata`

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scope
GET	<a href="#">Read Metadata of Events Aggregate [page 1182]</a>	/EventsAggregate/v1/\$metadata	<tde>.r
GET	<a href="#">Read Aggregate of Events [page 1183]</a>	/EventsAggregate/v1/Events?timeRange=<fromTime>-<toTime>&divisions=<division>	<tde>.r

#### i Note

<tde> refers to the event application name.

## 21.8.1 Read Metadata of Events Aggregate

With this method, you can retrieve the metadata of time division based aggregate of events.

### Request

**URI:** `/EventsAggregate/v1/$metadata`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `<td>.r`

### Request Parameters

None

### Request Example

`/EventsAggregate/v1/$metadata`

### Response

#### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

### Payload

**Format:** [XML](#)

Media types supported are [JSON](#) and [XML](#). The [XML](#) for this method has the following structure:

```
<EntityType Name="Events">
  <Key>
    <PropertyRef Name="EventType"></PropertyRef>
  </Key>
  <Property Name="EventType" Type="Edm.String" Nullable="false"></
Property>
  <Property Name="FirstOccuringEventId" Type="Edm.String"></
Property>
  <Property Name="Timestamp" Type="Edm.DateTimeOffset"></Property>
  <Property Name="Count" Type="Edm.Int32"></Property>
  <Property Name="ThingProperty" Type="Edm.String"></Property>
  <Property Name="ThingId" Type="Edm.String"></Property>
  <Property Name="SecondsDuration" Type="Edm.Double"></Property>
  <Property Name="Type" Type="Edm.String"></Property>
```

```

        <Property Name="CreatedBusinessTimestamp"
Type="Edm.DateTimeOffset"></Property>
    </EntityType>
    <EntityContainer Name="EventsAggregate"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="Events" EntityType="com.sap.appiot.esa.Event"></
EntitySet>
    </EntityContainer>

```

## Related Information

[Events Aggregate: OData Service \[page 1181\]](#)

## 21.8.2 Read Aggregate of Events

Retrieves aggregate of events grouped by the event type for the specified time interval

You can use this method to retrieve aggregates of events for each event type. The method performs the following:

- Retrieves latest events grouped by the correlation ID.
- Retrieves the first occurrence of the event per event type for the calculated time bucket. The method also retrieves the count events retrieved.

For example, consider the following set of events:

Event ID	Correlation ID	Timestamp	Event Type
E1	C1	10:00 am	ET1
E2	C1	10:10 am	ET1
E3	C1	11:10 am	ET1
E4	C2	09:00 am	ET1
E5	C2	09:30 am	ET1
E6	C3	10:10 am	ET2

The method `/EventsAggregate/v1/Events?`

`timerange=2016-02-10T09:00:00.514Z-2017-08-16T12:00:00.533Z&divisions=40` when executed, performs the following:

- Retrieves latest events grouped by the correlation ID as depicted in the following table:

Event ID	Correlation ID	Timestamp	Event Type
E3	C1	11:10 am	ET1
E5	C2	09:30 am	ET1

Event ID	Correlation ID	Timestamp	Event Type
E6	C3	10:10 am	ET2

- From the list of latest events grouped by the correlation ID, the method retrieves the first occurrence of the event per event type. In this example, event E5 of event type ET1 with correlation ID C2 and event E6 of event type ET2 with correlation ID C3.
- From the complete list of events, for a combination of event type (ET1) and correlation ID (C2), the method retrieves the first occurrence of the event (E4). The timestamp of this first occurred event (E4) created for the correlation ID (C2) is displayed as the created business timestamp in the response payload as indicated in the following table:

Event ID	Correlation ID	Timestamp	Event Type	Creation Time
E5	C2	09:30 am	ET1	09:00 am
E6	C3	10:10 am	ET2	10:10 am

## Request

**URI:** /EventsAggregate/v1/Events?timerange=<fromTime>-<toTime>&divisions=<division>

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** <td>.r

## Request Parameters

Parameter	Required	Data Type	Description
fromTime	Yes	Timestamp	Time from which the aggregates are calculated for events
toTime	Yes	Timestamp	Time up to which the aggregates are calculated for events
divisions	Yes	Integer	Number of divisions
thing ID	No	String	Unique identifier of a thing
thing property	No	String	Free text that describes the thing property
type	No	String	Free text that describes the event

## Query Request Parameters

Parameter	Required	Data Type	Description
\$filter	No	String	Filter condition to be applied; valid fields are <a href="#">ThingId</a> , <a href="#">ThingProperty</a> , and <a href="#">Type</a> .

## Request Example

```
/EventsAggregate/v1/Events?  
timerange=2016-02-10T13:10:16.514Z-2017-08-16T13:10:16.533Z&divisions=40&  
$filter=ThingId eq '0089A1D3E6C444E8894FEE517320EC92'
```

## Response

### Response Status and Error Codes

Code	Description
200	Data retrieved successfully

## Payload

Format: [XML](#)

Media types supported are [JSON](#) and [XML](#). The [XML](#) for this method has the following structure:

```
<entry>  
  <id>https://events-aggregate-sap-ci.cfapps.sap.hana.ondemand.com:443/  
EventsAggregate/v1/Events('core.automobiles%3AAlertEventType')</id>  
  <title type="text">Events</title>  
  <updated>2019-05-31T09:45:29.613Z</updated>  
  <category term="com.sap.apptot.esa.Events" scheme="http://  
schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category>  
  <link href="Events('core.automobiles%3AAlertEventType')" rel="edit"  
title="Events"></link>  
  <content type="application/xml">  
    <m:properties>  
      <d:EventType>core.automobiles:AlertEventType</d:EventType>  
      <d:FirstOccuringEventId>42FFF4BFB4844A77B19B0F3ED13A459D</  
d:FirstOccuringEventId>  
      <d:Timestamp>2018-07-08T05:30:00Z</d:Timestamp>  
      <d:Count>2</d:Count>  
      <d:ThingProperty>core.automobiles:TTCar/carWheel</  
d:ThingProperty>  
      <d:ThingId>DFA5E70345E24205A065939A59A47FF2</d:ThingId>  
      <d:SecondsDuration>7583400.0</d:SecondsDuration>  
      <d:Type m:null="true"></d:Type>  
      <d:CreatedBusinessTimestamp>2018-07-08T05:00:00Z</  
d:CreatedBusinessTimestamp>  
    </m:properties>  
  </content>  
</entry>  
<entry>  
  <id>https://events-aggregate-sap-ci.cfapps.sap.hana.ondemand.com:443/  
EventsAggregate/v1/Events('core.automobiles%3AAlertEventType1')</id>  
  <title type="text">Events</title>  
  <updated>2019-05-31T09:45:29.613Z</updated>  
  <category term="com.sap.apptot.esa.Events" scheme="http://  
schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category>  
  <link href="Events('core.automobiles%3AAlertEventType1')" rel="edit"  
title="Events"></link>  
  <content type="application/xml">  
    <m:properties>
```

```

        <d:EventType>core.automobiles:AlertEventType1</d:EventType>
        <d:FirstOccuringEventId>96003CD12C5242A9848967DCA3D15670</
d:FirstOccuringEventId>
        <d:Timestamp>2018-07-08T06:40:00Z</d:Timestamp>
        <d:Count>2</d:Count>
        <d:ThingProperty>core.automobiles:TTCar/carWheel</
d:ThingProperty>
        <d:ThingId>DFA5E70345E24205A065939A59A47FF2</d:ThingId>
        <d:SecondsDuration>7579200.0</d:SecondsDuration>
        <d:Type m:null="true"></d:Type>
        <d:CreatedBusinessTimestamp>2018-07-08T06:30:00Z</
d:CreatedBusinessTimestamp>
        </m:properties>
    </content>
</entry>

```

## Related Information

[Events Aggregate: OData Service \[page 1181\]](#)

## 22 Query Modeler Dynamic OData services

A query model has an OData service that is dynamically created in order to read the Facts and Things with their respective dimension values as defined in the model. Each service has the two entity sets:

- Facts
- Things

But, each entity set has a different set of properties based on the respective model.

Facts entity represent the aggregated values from several measurements. Things entity represent the collection of things that have aggregate values, which get included in the query model, and only the dimensions that were selected for the query model are the properties of the Thing entity. For example, if a Thing entity has properties such as color and size, but only color was selected by the user for the query model. In that case only color becomes a property of the Things entity in the dynamic OData service of the query model.

### i Note

Make sure you have a valid client credential token or one of the role collections required to use the respective query models.

**Resource Path:** `https://<query modeler url>/odata/v4/<tenant id>/<query model id>/v1`

### Operations

#### CRUD Operations

HTTP Method	Operation	Path
GET	<a href="#">Retrieve the List of Available Services [page 1188]</a>	<code>/odata/v4/\$list</code>
GET	<a href="#">Retrieve the List of Entity Sets of the Service [page 1189]</a>	<code>/odata/v4/&lt;tenant id&gt;/&lt;query model id&gt;/v1</code>
GET	<a href="#">Retrieve the Metadata of the Service [page 1191]</a>	<code>/odata/v4/&lt;tenant id&gt;/&lt;query model id&gt;/v1/\$metadata</code>
GET	<a href="#">Retrieve Personal Data about Data Subjects [page 1194]</a>	<code>/odata/v4/&lt;tenant id&gt;/&lt;query model id&gt;/v1/\$personal</code>
GET	<a href="#">Retrieve a Thing [page 1196]</a>	<code>/odata/v4/&lt;tenant id&gt;/&lt;query model id&gt;/v1/Things('&lt;thing id&gt;')</code>
GET	<a href="#">Retrieve the List of Things [page 1197]</a>	<code>/odata/v4/&lt;tenant id&gt;/&lt;query model id&gt;/v1/Things</code>
DELETE	<a href="#">Delete a Thing [page 1199]</a>	<code>/odata/v4/&lt;tenant id&gt;/&lt;query model id&gt;/v1/Things('&lt;thing id&gt;')</code>
GET	<a href="#">Retrieve a Fact [page 1200]</a>	<code>/odata/v4/&lt;tenant id&gt;/&lt;query model id&gt;/v1/Facts(ThingID='&lt;thing id&gt;', PointInTime=&lt;Edm.DateTimeOffset&gt;)</code>

HTTP Method	Operation	Path
<i>GET</i>	<a href="#">Retrieve the List of Facts [page 1201]</a>	/odata/v4/<tenant id>/<query model id>/v1/Facts
<i>DELETE</i>	<a href="#">Delete a Fact [page 1203]</a>	/odata/v4/<tenant id>/<query model id>/v1/Facts(ThingID='<thing id>', PointInTime=<Edm.DateTimeOffset>)

**OData Version: 4.0**

### Support of OData Features

Feature	Support
Query options	<p>Query Modeler service supports the following query options, which can be passed as query or path parameters:</p> <ul style="list-style-type: none"> <li>\$filter</li> <li>\$orderby (default for "Facts": PointInTime desc)</li> <li>\$top (default: 100, max: 1000)</li> <li>\$skip</li> <li>\$count</li> <li>\$expand</li> <li>\$select</li> </ul>
Exception handling	Exceptions are detected and logged on the top level of the service. The error messages are displayed as OData responses.

## 22.1 Services

### 22.1.1 Retrieve the List of Available Services

This method allows you to retrieve the list of available services.

**Base URI:** `https://<query modeler url>/odata/v4/$list`

### Request

**URI:** `/odata/v4/$list`

**HTTP:** *GET*

For more information on Roles and Permissions, see <https://help.sap.com/viewer/e057ad687acc4d2d8f2893609aff248b/1906a/en-US/26d64ee1664844849da6c5fe67240c90.html>.



## Request Parameters

None.

## Request Example

```
/odata/v4/$list
```

## Response

Format: **JSON**

### Sample Response:

#### Output Code

```
{
  "<query model id>": "/odata/v4/<tenant id>/<query model id>/v1"
}
```

### Response Status and Error Codes

Code	Description
200	Retrieved the list of available services successfully
400	Invalid request. Wrong format or structure of the provided request.
401	Unauthorized
403	Access denied. You did not have the required permissions to access the resource.
404	Service not found
405	Method not allowed
500	Internal server error

## 22.1.2 Retrieve the List of Entity Sets of the Service

This method allows you to retrieve the list of entity sets of the service

**Base URI:** `https://<query modeler url>/odata/v4/<tenant id>/<query model id>/v1`

## Request

**URI:** /odata/v4/<tenant id>/<query model id>/v1

**HTTP:** *GET*

For more information on Roles and Permissions, see <https://help.sap.com/viewer/e057ad687acc4d2d8f2893609aff248b/1906a/en-US/26d64ee1664844849da6c5fe67240c90.html>.

### Request Header

Header	Required	Description
Authorization	Yes	JWT (JSON Web Token)

### Request Parameters

Request Parameters

Name	Required	Data Type	Description
Content-Type	Optional	string	Entity sets JSON content

### Request Example

```
/odata/v4/<tenant id>/<query model id>/v1
```

## Response

Format: **JSON**

**Sample Response:**

≡ Output Code

```
{
  "@odata.context": "$metadata",
  "@odata.metadataEtag": "W/\\"zAoC82U0TeeFBtJIgBRpcI6dVUFjKoWJGEIR6vtkP/M=\\\"",
  "value": [
    {
      "name": "Things",
      "url": "Things"
    },
    {
      "name": "Facts",
      "url": "Facts"
    }
  ]
}
```

## Response Status and Error Codes

Code	Description
200	Retrieved the list of entity sets of a service successfully
400	Invalid request. Wrong format or structure of the provided request.
401	Unauthorized
403	Access denied. You did not have the required permissions to access the resource.
404	Entity sets of a service not found. Possible reasons: <ul style="list-style-type: none"><li>• You provided a wrong URL</li><li>• The given task you are referring to does not exist</li><li>• You are not allowed to access the task</li></ul>
405	Method not allowed
500	Internal server error

## 22.1.3 Retrieve the Metadata of the Service

This method allows you to retrieve the metadata of the service

**Base URI:** `https://<query modeler url>/odata/v4/<tenant id>/<query model id>/v1/$metadata`

### Request

**URI:** `/odata/v4/<tenant id>/<query model id>/v1/$metadata`

**HTTP:** *GET*

For more information on Roles and Permissions, see <https://help.sap.com/viewer/e057ad687acc4d2d8f2893609aff248b/1906a/en-US/26d64ee1664844849da6c5fe67240c90.html>.

### Request Header

Header	Required	Description
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Request Parameters

Name	Required	Data Type	Description
Content-Type	Optional	string	Metadata XML content

## Request Example

```
/odata/v4/<tenant id>/<query model id>/v1/$metadata
```

## Response

Format: **XML**

Sample Response:

### Output Code

```
<?xml version="1.0" encoding="utf-8"?>
<edmx:Edmx Version="4.0" xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx">
  <edmx:Reference Uri="https://oasis-tcs.github.io/odata-vocabularies/vocabularies/Org.OData.Core.V1.xml">
    <edmx:Include Alias="Core" Namespace="Org.OData.Core.V1"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://oasis-tcs.github.io/odata-vocabularies/vocabularies/Org.OData.Measures.V1.xml">
    <edmx:Include Alias="Measures" Namespace="Org.OData.Measures.V1"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://oasis-tcs.github.io/odata-vocabularies/vocabularies/Org.OData.Capabilities.V1.xml">
    <edmx:Include Alias="Capabilities"
    Namespace="Org.OData.Capabilities.V1"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://oasis-tcs.github.io/odata-vocabularies/vocabularies/Org.OData.Aggregation.V1.xml">
    <edmx:Include Alias="Aggregation"
    Namespace="Org.OData.Aggregation.V1"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://oasis-tcs.github.io/odata-vocabularies/vocabularies/Org.OData.Validation.V1.xml">
    <edmx:Include Alias="Validation" Namespace="Org.OData.Validation.V1"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://wiki.scn.sap.com/wiki/download/attachments/462030211/Analytics.xml?api=v2">
    <edmx:Include Alias="Analytics"
    Namespace="com.sap.vocabularies.Analytics.v1"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://wiki.scn.sap.com/wiki/download/attachments/448470974/Common.xml?api=v2">
    <edmx:Include Alias="Common"
    Namespace="com.sap.vocabularies.Common.v1"/>
  </edmx:Reference>
  <edmx:Reference Uri="https://wiki.scn.sap.com/wiki/download/attachments/448470971/Communication.xml?api=v2">
    <edmx:Include Alias="Communication"
    Namespace="com.sap.vocabularies.Communication.v1"/>
  </edmx:Reference>
```

```

    <edmx:Reference Uri="https://wiki.scn.sap.com/wiki/download/attachments/
448470968/UI.xml?api=v2">
      <edmx:Include Alias="UI" Namespace="com.sap.vocabularies.UI.v1"/>
    </edmx:Reference>
    <edmx:DataServices>
      <Schema Namespace="sap.iot.qm.gen.Test.Dynamic" xmlns="http://
docs.oasis-open.org/odata/ns/edm">
        <EntityContainer Name="EntityContainer">
          <EntitySet Name="Things"
EntityType="sap.iot.qm.gen.Test.Dynamic.Things"/>
          <EntitySet Name="Facts"
EntityType="sap.iot.qm.gen.Test.Dynamic.Facts"/>
        </EntityContainer>
        <EntityType Name="Things">
          <Key>
            <PropertyRef Name="ThingID"/>
          </Key>
          <Property Name="ThingID" Type="Edm.String" MaxLength="256"
Nullable="false"/>
          <Property Name="ThingDimensionOne" Type="Edm.Boolean"/>
          <Property Name="ThingDimensionTwo" Type="Edm.String"/>
        </EntityType>
        <EntityType Name="Facts">
          <Key>
            <PropertyRef Name="ThingID"/>
            <PropertyRef Name="PointInTime"/>
          </Key>
          <Property Name="ThingID" Type="Edm.String" MaxLength="256"
Nullable="false"/>
          <Property Name="PointInTime" Type="Edm.DateTimeOffset"
Nullable="false"/>
          <Property Name="FactOne" Type="Edm.Double"/>
          <Property Name="FactTwo" Type="Edm.Double"/>
        </EntityType>
        <Annotations Target="sap.iot.qm.gen.Test.Dynamic.EntityContainer/
Things">
          <Annotation Term="Capabilities.InsertRestrictions">
            <Record Type="Capabilities.InsertRestrictionsType">
              <PropertyValue Property="Insertable" Bool="false"/>
            </Record>
          </Annotation>
          <Annotation Term="Capabilities.UpdateRestrictions">
            <Record Type="Capabilities.UpdateRestrictionsType">
              <PropertyValue Property="Updatable" Bool="false"/>
            </Record>
          </Annotation>
        </Annotations>
        <Annotations Target="sap.iot.qm.gen.Test.Dynamic.EntityContainer/
Facts">
          <Annotation Term="Capabilities.InsertRestrictions">
            <Record Type="Capabilities.InsertRestrictionsType">
              <PropertyValue Property="Insertable" Bool="false"/>
            </Record>
          </Annotation>
          <Annotation Term="Capabilities.UpdateRestrictions">
            <Record Type="Capabilities.UpdateRestrictionsType">
              <PropertyValue Property="Updatable" Bool="false"/>
            </Record>
          </Annotation>
        </Annotations>
      </Schema>
    </edmx:DataServices>
  </edmx:Edmx>

```

## Response Status and Error Codes

Code	Description
200	Retrieved the metadata of the service successfully
400	Invalid request. Wrong format or structure of the provided request.
401	Unauthorized
403	Access denied. You did not have the required permissions to access the resource.
404	Metaservice of a service not found. Possible reasons: <ul style="list-style-type: none"><li>• You provided a wrong URL</li><li>• The given task you are referring to does not exist</li><li>• You are not allowed to access the task</li></ul>
405	Method not allowed
500	Internal server error

## 22.1.4 Retrieve Personal Data about Data Subjects

This method allows you to retrieve the list of personal and sensitive data of the data subjects that has been stored.

**Base URI:** `https://<query modeler url>/odata/v4/<tenant id>/<query model id>/v1/$personal`

### Request

**URI:** `/odata/v4/<tenant id>/<query model id>/v1/$personal`

**HTTP:** *GET*

For more information on Roles and Permissions, see <https://help.sap.com/viewer/e057ad687acc4d2d8f2893609aff248b/1906a/en-US/26d64ee1664844849da6c5fe67240c90.html>.

### Request Header

Header	Required	Description
Authorization	Yes	JWT (JSON Web Token)

## Request Parameters

Request Parameters

Name	Required	Data Type	Description
Content-Type	Optional	string	Personal XML content

## Request Example

```
/odata/v4/<tenant id>/<query model id>/v1/$personal
```

## Response

Format: **XML**

Sample Response:

≡ Output Code

```
{
  "Facts": {
    "@com.sap.apptot.security:pii": [],
    "@com.sap.apptot.security:spi": []
  },
  "Things": {
    "@com.sap.apptot.security:pii": [],
    "@com.sap.apptot.security:spi": [
      "ThingDimensionOne",
      "ThingDimensionTwo"
    ]
  }
}
```

Response Status and Error Codes

Code	Description
200	Retrieved the list of sensitive personal data successfully
400	Invalid request. Wrong format or structure of the provided request.
401	Unauthorized
403	Access denied. You did not have the required permissions to access the resource.
404	Personal sensitive data not found. Possible reasons: <ul style="list-style-type: none"><li>• You provided a wrong URL</li><li>• The given task you are referring to does not exist</li><li>• You are not allowed to access the task</li></ul>

Code	Description
405	Method not allowed
500	Internal server error

## 22.2 Things

### 22.2.1 Retrieve a Thing

This method allows you to retrieve a particular thing with it's respective dimensions.

**Base URI:** `https://<query modeler url>/odata/v4/<tenant id>/<query model id>/v1/Things('<thing id>')`

#### Request

**URI** `/odata/v4/<tenant id>/<query model id>/v1/Things('<thing id>')`

**HTTP:** [GET](#)

For more information on Roles and Permissions, see <https://help.sap.com/viewer/e057ad687acc4d2d8f2893609aff248b/1906a/en-US/26d64ee1664844849da6c5fe67240c90.html>.

#### Request Parameters

Name	Required	Data Type	Description
thing id	Yes	string	The ID of a Thing

#### Request Example

```
/odata/v4/<tenant id>/<query model id>/v1/Things('<thing id>')
```

#### Response

Format: **JSON**

**Sample Response for \$entity:**

⌵, Output Code

```
{
```



```

"@odata.context": "$metadata#Things/$entity",
"@odata.metadataEtag": "W/\\"zAoC82U0TeeFBtJIgBRpcI6dVUFjKoWJGEIR6vtpK/M=
\\"",
"ThingID": "1",
"ThingDimensionOne": true,
"ThingDimensionTwo": "TEST 1"
}

```

#### Response Status and Error Codes

Code	Description
200	Retrieved the thing successfully
400	Invalid request. Wrong format or structure of the provided request.
401	Unauthorized
403	Access denied. You did not have the required permissions to access the resource.
404	Thing not found.
405	Method not allowed
500	Internal server error

## 22.2.2 Retrieve the List of Things

This method allows you to retrieve the list of things and their respective dimension values.

**Base URI:** `https://<query modeler url>/odata/v4/<tenant id>/<query model id>/v1/Things`

### Request

**URI:** `/odata/v4/<tenant id>/<query model id>/v1/Things`

**HTTP:** `GET`

For more information on Roles and Permissions, see <https://help.sap.com/viewer/e057ad687acc4d2d8f2893609aff248b/1906a/en-US/26d64ee1664844849da6c5fe67240c90.html>.

### Request Parameters

None.

### Request Example

```
/odata/v4/<tenant id>/<query model id>/v1/Things
```

## Response

Format: **JSON**

**Sample Response:**

≡ Output Code

```
{
  "@odata.context": "$metadata#Things",
  "@odata.metadataEtag": "W/\\"zAoC82U0TeeFBtJIgBRpcI6dVUFjKoWJGEIR6vtkP/M=\\\"",
  "value": [
    {
      "ThingID": "1",
      "ThingDimensionOne": true,
      "ThingDimensionTwo": "TEST 1"
    },
    {
      "ThingID": "2",
      "ThingDimensionOne": true,
      "ThingDimensionTwo": "TEST 2"
    },
    {
      "ThingID": "3",
      "ThingDimensionOne": false,
      "ThingDimensionTwo": "TEST 3"
    }
  ]
}
```

Response Status and Error Codes

Code	Description
200	Retrieved the list of things successfully
400	Invalid request. Wrong format or structure of the provided request.
401	Unauthorized
403	Access denied. You did not have the required permissions to access the resource.
404	Resource not found
405	Method not allowed
500	Internal server error

## 22.2.3 Delete a Thing

This method allows you to delete a particular thing with it's respective dimensions.

**Base URI:** `https://<query modeler url>/odata/v4/<tenant id>/<query model id>/v1/Things('<thing id>')`

### Request

**URI** `/odata/v4/<tenant id>/<query model id>/v1/Things('<thing id>')`

**HTTP:** *DELETE*

For more information on Roles and Permissions, see <https://help.sap.com/viewer/e057ad687acc4d2d8f2893609aff248b/1906a/en-US/26d64ee1664844849da6c5fe67240c90.html>.

### Request Parameters

Name	Required	Data Type	Description
thing id	Yes	string	The ID of the Thing to be deleted

### Request Example

```
/odata/v4/<tenant id>/<query model id>/v1/Things('<thing id>')
```

### Response

Response Status and Error Codes

Code	Description
204	The Thing has been deleted successfully
400	Invalid request. Wrong format or structure of the provided request.
401	Unauthorized
403	Access denied. You did not have the required permissions to access the resource.
404	Thing not found.
405	Method not allowed

Code	Description
500	Internal server error. The operation you requested led to an error during execution.

## 22.3 Facts

### 22.3.1 Retrieve a Fact

This method allows you to retrieve a particular fact.

**Base URI:** `https://<query modeler url>/odata/v4/<tenant id>/<query model id>/v1/Facts(ThingID='<thing id>',PointInTime=<Edm.DateTimeOffset>)`

#### Request

**URI** `/odata/v4/<tenant id>/<query model id>/v1/Facts(ThingID='<thing id>',PointInTime=<Edm.DateTimeOffset>)`

**HTTP:** *GET*

For more information on Roles and Permissions, see <https://help.sap.com/viewer/e057ad687acc4d2d8f2893609aff248b/1906a/en-US/26d64ee1664844849da6c5fe67240c90.html>.

#### Request Parameters

Name	Required	Data Type	Description
thing id	Yes	string	Fact of a thing ID

#### Request Example

```
/odata/v4/<tenant id>/<query model id>/v1/Facts(ThingID='<thing id>',PointInTime=<Edm.DateTimeOffset>)
```

#### Response

Format: **JSON**

### Sample Response for \$entity:

#### Output Code

```
{
  "@odata.context": "$metadata#Facts/$entity",
  "@odata.metadataEtag": "W/\\"zAoC82U0TeeFBtJIgBRpcI6dVUFjKoWJGEIR6vtkP/M=\\\"",
  "ThingID": "1",
  "PointInTime": "2019-01-01T12:00:01.000Z",
  "FactOne": 1.1,
  "FactTwo": 1.2
}
```

#### Response Status and Error Codes

Code	Description
200	Retrieved the fact successfully
400	Invalid request. Wrong format or structure of the provided request.
401	Unauthorized
403	Access denied. You did not have the required permissions to access the resource.
404	Fact not found.
405	Method not allowed
500	Internal server error

## 22.3.2 Retrieve the List of Facts

This method allows you to retrieve the list of facts available.

**Base URI:** `https://<query modeler url>/odata/v4/<tenant id>/<query model id>/v1/Facts`

### Request

**URI** `/odata/v4/<tenant id>/<query model id>/v1/Facts`

**HTTP:** *GET*

For more information on Roles and Permissions, see <https://help.sap.com/viewer/e057ad687acc4d2d8f2893609aff248b/1906a/en-US/26d64ee1664844849da6c5fe67240c90.html>.

## Request Parameters

None.

## Request Example

```
/odata/v4/<tenant id>/<query model id>/v1/Facts
```

## Response

Format: **JSON**

### Sample Response:

#### Output Code

```
{
  "@odata.context": "$metadata#Facts",
  "@odata.metadataEtag": "W/\\"zAoC82U0TeeFBtJIgBRpcI6dVUFjKoWJGEIR6vtkP/M=\\\"",
  "value": [
    {
      "ThingID": "2",
      "PointInTime": "2019-01-01T12:00:02.000Z",
      "FactOne": 2.1,
      "FactTwo": 2.2
    },
    {
      "ThingID": "3",
      "PointInTime": "2019-01-01T12:00:01.000Z",
      "FactOne": 3.1,
      "FactTwo": 3.2
    },
    {
      "ThingID": "1",
      "PointInTime": "2019-01-01T12:00:01.000Z",
      "FactOne": 1.1,
      "FactTwo": 1.2
    }
  ]
}
```

### Response Status and Error Codes

Code	Description
200	Retrieved the list of facts successfully
400	Invalid request. Wrong format or structure of the provided request.
401	Unauthorized
403	Access denied. You did not have the required permissions to access the resource.

Code	Description
404	Fact not found.
405	Method not allowed
500	Internal server error

## 22.3.3 Delete a Fact

This method allows you to delete a particular fact.

**Base URI:** `https://<query modeler url>/odata/v4/<tenant id>/<query model id>/v1/Facts(ThingID='<thing id>', PointInTime=<Edm.DateTimeOffset>)`

### Request

**URI** `/odata/v4/<tenant id>/<query model id>/v1/Facts(ThingID='<thing id>', PointInTime=<Edm.DateTimeOffset>)`

**HTTP:** *DELETE*

For more information on Roles and Permissions, see <https://help.sap.com/viewer/e057ad687acc4d2d8f2893609aff248b/1906a/en-US/26d64ee1664844849da6c5fe67240c90.html>.

### Request Parameters

Name	Required	Data Type	Description
thing id	Yes	string	The fact of a thing ID to be deleted

### Request Example

```
/odata/v4/<tenant id>/<query model id>/v1/Facts(ThingID='<thing id>', PointInTime=<Edm.DateTimeOffset>)
```

## Response

### Response Status and Error Codes

Code	Description
204	The fact has been deleted successfully
400	Invalid request. Wrong format or structure of the provided request.
401	Unauthorized
403	Access denied. You did not have the required permissions to access the resource.
404	Fact not found.
405	Method not allowed
500	Internal server error. The operation you requested led to an error during execution.



# 23 Monitoring and Logging

## Overview

Using the monitoring and the logging service, you can retrieve the status of a service executed. For example, the service to create multiple things generates a correlation ID.

- You can use the monitoring service to retrieve status of correlation IDs. The response of the monitoring service includes statuses such as **Success**, **In Process**, **Error**, or **Partial Success** and a detailed log for the service executed.
- You can use the logging service to retrieve the log details of a specific correlation ID and in turn the details of individual logs.

**Resource Path:** `http://<server address>[:<port number>][/path]/ServiceMonitoring`

## Operations

### CRUD Operations

HTTP Method	Operation	URI	Scopes
GET	<a href="#">Read all Correlations [page 1205]</a>	/Logging/Correlations	monitor.r
GET	<a href="#">Read Status of a Correlation ID [page 1208]</a>	/State/Status('<<Correlation ID>>')	monitor.r
GET	<a href="#">Read Log Details of a Correlation [page 1210]</a>	/Logging/Correlations('<<Correlation ID>>')/Logs	monitor.r
GET	<a href="#">Read Details of a Log [page 1212]</a>	/Logging/Logs('<<log ID>>')	monitor.r

## 23.1 Read all Correlations

With this method, you can retrieve all correlation IDs. The response includes the following details:

- Correlation ID
- Timestamp
- Number of requests processed, for example, the number of thing instances to be created provided in the request payload
- Number of requests succeeded, for example, the number of thing instances created successfully
- Number of requests failed, for example, the number of thing instances failed to be created

## Request

URI: /Logging/Correlations

Operation Type: CRUD

HTTP Method: [GET](#)

Permissions: monitor.r

### Request Parameters

None

### Query String Parameters

Parameter	Required	Data Type	Description
\$filter	No	String	Filter condition to be applied; the only valid field is correlation ID. The filter operators supported are eq, lt, gt, le, ge, and ne.  Multiple filters are supported using the separators – and, or.

### Request Example

/Logging/Correlations

## Response

### Response Status and Error Codes

Code	Description
200	Correlation details are retrieved successfully

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "results": [{
      "__metadata": {
```

```

        "id": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Correlations('BF517F7DB459490CB03D71C9A28B5822')",
        "uri": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Correlations('BF517F7DB459490CB03D71C9A28B5822')",
        "type": "com.sap.apptot.service.Correlation"
    },
    "CorrelationID": "BF517F7DB459490CB03D71C9A28B5822",
    "ProcessedRequestsCount": 3,
    "RequestFailureCount": 0,
    "RequestSuccessCount": 3,
    "Logs": {
        "__deferred": {
            "uri": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Correlations('BF517F7DB459490CB03D71C9A28B5822')/Logs"
        }
    }
},
{
    "__metadata": {
        "id": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Correlations('E7F632BF57D04F0CA9C531863364A92D')",
        "uri": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Correlations('E7F632BF57D04F0CA9C531863364A92D')",
        "type": "com.sap.apptot.service.Correlation"
    },
    "CorrelationID": "E7F632BF57D04F0CA9C531863364A92D",
    "ProcessedRequestsCount": 1,
    "RequestFailureCount": 1,
    "RequestSuccessCount": 0,
    "Logs": {
        "__deferred": {
            "uri": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Correlations('E7F632BF57D04F0CA9C531863364A92D')/Logs"
        }
    }
},
{
    "__metadata": {
        "id": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Correlations('04FBF4A2DD34456E87027CBEB24AC407')",
        "uri": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Correlations('04FBF4A2DD34456E87027CBEB24AC407')",
        "type": "com.sap.apptot.service.Correlation"
    },
    "CorrelationID": "04FBF4A2DD34456E87027CBEB24AC407",
    "ProcessedRequestsCount": 1,
    "RequestFailureCount": 1,
    "RequestSuccessCount": 0,
    "Logs": {
        "__deferred": {
            "uri": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Correlations('04FBF4A2DD34456E87027CBEB24AC407')/Logs"
        }
    }
}
]
}
}

```

## Related Information

[Monitoring and Logging \[page 1205\]](#)

## 23.2 Read Status of a Correlation ID

With this method, you can read the status of a specific correlation ID.

The response includes a correlation ID along with its corresponding timestamp, service name, and status. The following values indicate different statuses of the service:

- **Success** – The service successfully completed. For example, if the service is executed to create three thing instances, this status indicates that all three thing instances are created successfully.
- **In Process** – The service is in process. For example, if the service is executed to create 30 thing instances, this status indicated that creation of some of the thing instances is in progress.
- **Error** – The service failed. For example, if the service is executed to create three thing instances, this status indicates that none of the thing instances are created.
- **Partial Success** – The service partially succeeded. For example, if the service is executed to create three thing instances, the status indicates that one or more thing instances are not created successfully.

## Request

**URI:** `/State/Status('<<Correlation ID>>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `monitor.r`

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
correlation ID	Yes	GUID	Indicates the unique correlation ID	Path

## Query String Parameters

Parameter	Required	Data Type	Description
\$filter	No	String	<p>Filter condition to be applied; the valid fields are correlation ID, timestamp, service name, and status. The filter operators supported are eq, lt, gt, le, ge, and ne.</p> <p>Multiple filters are supported using the separators – and, or.</p>

## Request Example

```
/State/Status('F8E69835B61F4130B12DB342043DA116')
```

## Response

### Response Status and Error Codes

Code	Description
200	Status of a specific correlation ID retrieved successfully

## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "__metadata": {
      "id": "https://service-metrics-sap-ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/State/Status('F8E69835B61F4130B12DB342043DA116')",
      "uri": "https://service-metrics-sap-ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/State/Status('F8E69835B61F4130B12DB342043DA116')",
      "type": "com.sap.apptot.service.Status"
    },
    "CorrelationID": "F8E69835B61F4130B12DB342043DA116",
    "Timestamp": "/Date(1557113529515)/",
    "ServiceName": "/v2/Things",
    "Status": "Success"
  }
}
```

## Related Information

[Monitoring and Logging \[page 1205\]](#)

### 23.3 Read Log Details of a Correlation

With this method, you can retrieve the status of the individual requests corresponding to a correlation ID. For example, the individual status of the thing instances to be created specified in the request payload. For every single thing instance creation request, a log ID is generated. The response includes the following details:

- Log ID, for example, unique log IDs for every thing instance to be created.
- Timestamp
- Log Status, for example, whether the creation of thing instance corresponding to this log ID succeeded or failed.

#### Request

**URI:** /Logging/Correlations('<<Correlation ID>>')/Logs

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** monitor.r

#### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
correlation ID	Yes	GUID	Unique correlation ID	Path

#### Query String Parameters

Parameter	Required	Data Type	Description
\$filter	No	String	Filter condition to be applied; the valid fields are log ID and log status. The filter operators supported are eq, lt, gt, le, ge, and ne.  Multiple filters are supported using the separators – and, or.

## Request Example

/Logging/Correlations('<<Correlation ID>>')/Logs

## Response

### Response Status and Error Codes

Code	Description
200	Log details retrieved successfully

## Payload

Format: *JSON*

Media types supported are *JSON* and *XML*. The *JSON* for this method has the following structure:

```
{
  "d": {
    "results": [
      {
        "__metadata": {
          "id": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Logs('C4244252A7EA43F892B2FE82CC7C64B6')",
          "uri": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Logs('C4244252A7EA43F892B2FE82CC7C64B6')",
          "type": "com.sap.apptot.service.Log"
        },
        "LogID": "C4244252A7EA43F892B2FE82CC7C64B6",
        "Timestamp": "/Date(1557734655033)/",
        "LogStatus": "Success",
        "LogDetails": null
      },
      {
        "__metadata": {
          "id": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Logs('66FA474E6BE54D568571C70A4C6EED8B')",
          "uri": "https://service-metrics-sap-
ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/
Logs('66FA474E6BE54D568571C70A4C6EED8B')",
          "type": "com.sap.apptot.service.Log"
        },
        "LogID": "66FA474E6BE54D568571C70A4C6EED8B",
        "Timestamp": "/Date(1557734655033)/",
        "LogStatus": "Success",
        "LogDetails": null
      }
    ]
  }
}
```

## Related Information

[Monitoring and Logging \[page 1205\]](#)

## 23.4 Read Details of a Log

With this method, you can read the details of a specific log ID. For example, whether the creation of thing instance corresponding to the log ID succeeded or failed. The response includes the following details:

- Log ID
- Timestamp
- Log Status, for example, Success or Error status with creation of a specific thing instance.
- Log Details, for example, the message text on the status of the thing creation. This includes the thing ID and thing name.

### Request

**URI:** `/Logging/Logs('<<log ID>>')`

**Operation Type:** CRUD

**HTTP Method:** [GET](#)

**Permissions:** `monitor.r`

### Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
log ID	Yes	GUID	Unique log ID	Path

### Request Example

`/Logging/Logs('C4244252A7EA43F892B2FE82CC7C64B6')`

### Response

#### Response Status and Error Codes

Code	Description
200	Logs details retrieved successfully



## Payload

Format: [JSON](#)

Media types supported are [JSON](#) and [XML](#). The [JSON](#) for this method has the following structure:

```
{
  "d": {
    "__metadata": {
      "id": "https://service-metrics-sap-ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/Logs('C4244252A7EA43F892B2FE82CC7C64B6')",
      "uri": "https://service-metrics-sap-ci.cfapps.sap.hana.ondemand.com:443/ServiceMonitoring/Logging/Logs('C4244252A7EA43F892B2FE82CC7C64B6')",
      "type": "com.sap.apptot.service.Log"
    },
    "LogID": "C4244252A7EA43F892B2FE82CC7C64B6",
    "Timestamp": "/Date(1557734655033)/",
    "LogStatus": "Success",
    "LogDetails": "{ \"thingId\": \"B739F3004387424A9CAFB832C9B52642\", \"name\": \"BMWCarX2\", \"url\": \"https://apptot-mds-ci.cfapps.sap.hana.ondemand.com/Things('B739F3004387424A9CAFB832C9B52642')\", \"alternateId\": \"BMWCarX2-altid\" }"
  }
}
```

## Related Information

[Monitoring and Logging \[page 1205\]](#)

## 24 Filter Conditions (\$filter)

You can restrict the data to be retrieved by using filter conditions. The filter conditions are validated before being applied.

### Overview

All services that are available in the SAP Leonardo IoT platform offer a method to retrieve a collection of either all objects or a subset of objects of a certain type from the database. This kind of method is often referred to as a "collective GET request". In this documentation, you can find these methods by their topic title always starting with *Read all <objects>*, for example, *Read all Persons*. By default, all these methods come with a filter parameter that allows for defining simple as well as more advanced object selection criteria. Although the standard filter may be simple, it is still important and often even required to avoid excessive server load caused by reading too much data from the server.

#### i Note

Filter conditions can be defined for many properties, but **not for all** the properties in the payload of a particular service. You can find a list of the fields supported for creating filter conditions in the documentation of each individual *Read All* service method.

### Availability

#### i Note

The filter conditions described in this section are currently available for the following services:

Services With Filter Support

Area	Services
Configuration	thingType
	propertySetType
Thing	All services
Thing Hierarchy	All services
Event, Event Type	All services

Area	Services
Authorization	ObjectGroup
	ObjectGroupAssignment
	ObjectGroupCollection
	ObjectGroupCollectionItem
	Capability
Location	All services
Value Helps	All services

## Details

Parameter: `$filter`

The format of valid filter conditions is as follows:

- Single filter condition:

```
<filter_property_name> <filter_operator> <filter_operand>
```

- Multiple filter conditions:

```
<filter_property_name> <filter_operator> <filter_operand> and
<filter_property_name> <filter_operator> <filter_operand>
```

Filter Parameter	Values														
<code>&lt;filter_property_name&gt;</code>	Any of the properties on which filter condition is allowed. Any invalid property name is a bad request.														
<code>&lt;filter_operator&gt;</code>	<p>The following binary filter operators are supported:</p> <table> <tr> <th>Operator</th><th>Description</th></tr> <tr> <td>eq</td><td>is equal</td></tr> <tr> <td>ne</td><td>is not equal</td></tr> <tr> <td>gt</td><td>is greater than</td></tr> <tr> <td>ge</td><td>is greater than or equal</td></tr> <tr> <td>lt</td><td>is less than</td></tr> <tr> <td>le</td><td>is less than or equal</td></tr> </table>	Operator	Description	eq	is equal	ne	is not equal	gt	is greater than	ge	is greater than or equal	lt	is less than	le	is less than or equal
Operator	Description														
eq	is equal														
ne	is not equal														
gt	is greater than														
ge	is greater than or equal														
lt	is less than														
le	is less than or equal														
<code>&lt;filter_operand&gt;</code>	Valid operands are <i>string</i> , <i>integer</i> , <i>float</i> , <i>true</i> , <i>false</i> , <i>null</i> , <i>date</i> , <i>datetime</i> , and <i>timestamp</i> .														

In a single method call, you can combine any number of filter conditions with the operators listed above with the `and` or the `or` operator. Any other join condition is a bad request.

The filter parameters are validated as follows:

- If `<filter_operand>` is null, the only supported operators are `eq` and `ne`.
- The `<filter_operand>` of type string is always enclosed within single quotes and the only special characters not supported are forward slash (/) and caret (^).
- The `<filter_operand>` of type `integer`, `float`, `null`, `true`, `false`, `date`, `datetime`, and `timestamp` are not enclosed within quotes. If these are enclosed within single quotes, it is treated as string.
- The `<filter_operand>` cannot contain special characters ampersand (&) and pound character (#). For example, `$filter=myString eq 'something1&something2'` is an invalid filter condition.
  - The special character & is used to separate query parameters, for example, `?query1=value&query2=value`.
  - The special character # is used as an anchor tag in URLs to navigate directly to a specific section in an html page or a document, for example, `http://<baseURL>/<resourcePath>#<element>`.
  - However, you can use `%26` instead of & and `%23` instead of #. For example, `$filter=myString eq 'something1%26something2'` is a valid filter condition.
- The type of `<filter_property_name>` and `<filter_operand>` should be the same.

#### Note

- Any filter condition in an invalid format is a bad request.
- A filter condition without the filter parameters such as filter property, filter operator, and filter operand in the request is not supported. For example, `/Things?$orderby=_thingID asc&$filter=()` is an invalid request.
- Data types `JSON`, `ByteArray`, and `LargeString` are not supported in `$filter`.
- The filter operator `eq` in combination with filter operand `true` is not mandatory in a filter condition. For example, the filter conditions `substringof(<'search string'>,<string field>) eq 'true'` and `substringof(<'search string'>,<string field>)` are valid. However, if you opt to use the filter operator `eq` with the filter operand `true`, the representation is supported as follows:
  - In `OData` service, you must not specify the filter operand `true` within quotes. For example, `substringof(<'search string'>,<string field>) eq 'true'` is invalid.

## Grouping Multiple Filter Conditions

You can define multiple filter conditions using the separators `and` and `or`. You can use the parenthesis to group the filter conditions to execute them in a particular order of sequence. This way, the result set gets well defined.

For example, the query with filter condition `/Things?$top=50&$filter=_id eq '327036AE27D94E9B9B85FC91A3ED0BC0' and _thingType eq 'core.automobiles:ABC2Series' or _name eq 'ABC2Series'` is executed as follows:

- **Result 1:** Retrieve a thing with ID '327036AE27D94E9B9B85FC91A3ED0BC0' and thing type name 'core.automobiles:ABC2Series'.
- **Result 2:** Retrieve all things with name 'ABC2Series'.
- **Result 3:** Result 1 or Result 2.

On the other hand, the query with filter condition `/Things?$top=50&$filter=_id eq '327036AE27D94E9B9B85FC91A3ED0BC0' and (_thingType eq 'core.automobiles:ABC2Series' or _name eq 'ABC2Series') eq true` is executed as follows:

- **Result 1:** Retrieve a thing with thing type name `'core.automobiles:ABC2Series'` or with name `'ABC2Series'`.
- **Result 2:** Retrieve a thing with ID `'327036AE27D94E9B9B85FC91A3ED0BC0'`.
- **Result 3:** Result 1 and Result 2.

## Examples

```
/Things?$filter=_thingType eq 'core.automobiles:ABC2Series' and _location ne DE
```

Retrieves all things that belong to the thing type `core.automobiles:ABC2Series` but not located in Germany.

```
/Things?$filter=( _thingType eq 'core.automobiles:ABC2Series' or _name eq 'ABC2Series') eq true) and Capacity lt 1000
```

- Firstly, retrieves all things that belong to the thing type `core.automobiles:ABC2Series` or with the name `'ABC2Series'`.
- From the result set, retrieves only those things with the property `Capacity` less than 1000.

```
/Things?$filter=((_thingType eq 'core.automobiles:ABC2Series' or _thingType eq 'core.automobiles:ABCSeries') or _name eq 'ABC2Series') eq true
```

- Retrieves all things that belong to the thing type `core.automobiles:ABC2Series` or `core.automobiles:ABCSeries`
- If there are no things found, the system retrieves all things with name `'ABC2Series'`

```
/Things/?$filter=_thingType ne 'core.automobiles:ABC2Series' and thingType ne 'core.automobiles:ABCSeries'
```

Retrieves all things that does not belong to thing types `core.automobiles:ABC2Series` and `core.automobiles:ABCSeries`.

## Search String Data Using \$filter

You can use string functions such as [substringof](#), [startswith](#), and [endswith](#) in filter conditions to search data of type string. You can filter data using a combination of these functions, for example `/Configuration/ThingTypes?$filter=substringof(<'search string'>,<string field>) eq 'true' or startswith(<string field>,<'search string'>) eq 'true'` is a valid filter condition.

### Note

Data search using string functions such as [substringof](#), [startswith](#), and [endswith](#) in `$filter` condition is not case-sensitive. For example, the requests `/Configuration/ThingTypes?$filter=startswith(Name,'ABC')` and `/Configuration/ThingTypes?$filter=startswith(Name,'abc')` retrieves identical results.

Following are a few examples of valid filter conditions using these string functions:

- /Configuration/ThingTypes?\$filter=startswith(<string field>,<'search string'>)
- /Configuration/ThingTypes?\$filter=startswith(<string field>,<'search string'>) eq 'true' and endswith(<string field>,<'search string'>) eq 'true'
- /Configuration/ThingTypes?\$filter=substringof(<'search string'>,<string field>) eq 'true' or endswith(<string field>,<'search string'>) eq 'true'
- /Configuration/ThingTypes?\$filter=substringof(<'search string'>,<string field1>) eq 'true' or substringof(<'search string'>,<string field2>) eq 'true'
- /Configuration/ThingTypes?\$filter= startswith(<string field1>,<'search string'>) eq 'true' or endswith(<string field2>,<'search string'>) eq 'true'
- /Configuration/ThingTypes?\$filter= startswith(<string field1>,<'search string'>) eq 'true' and endswith(<string field2>,<'search string'>) eq 'true' and substringof(<'search string'>,<string field3>)

## Filter Data Using Date Functions

You can use date functions such as [day](#), [hour](#), [minute](#), [second](#), [month](#), and [year](#) in filter conditions to search data. These functions are supported only for the property type of data type [DateTime](#). You can use the date functions only in the following APIs:

- /CompositeEvents/v1/<'property set type name'>?\$filter=Path eq <'property set ID'>
- /CompositeEvents/v2/<'property set type name'>?\$filter=Path eq <'property set ID'>
- /CompositeThings/v1/'<property set type name>'?\$filter=Path eq '<property set ID>'

Following are a few examples of valid filter conditions using these date functions:

- /CompositeEvents/v2/PST\_com\_abc\_core\_\_engine?\$filter=Path eq 'engine\_temp' and At\_DateTime eq datetime'2018-09-10T10:15:40'
- /CompositeEvents/v2/PST\_com\_abc\_core\_\_engine?\$filter=Path eq 'engine\_temp' and day(At\_DateTime) eq 10
- /CompositeEvents/v2/PST\_com\_abc\_core\_\_engine?\$filter=Path eq 'engine\_temp' and month(At\_DateTime) eq 9
- /CompositeEvents/v2/PST\_com\_abc\_core\_\_engine?\$filter=Path eq 'engine\_temp' and year(At\_DateTime) eq 2018
- /CompositeEvents/v2/PST\_com\_abc\_core\_\_engine?\$filter=Path eq 'engine\_temp' and hour(At\_DateTime) eq 10
- /CompositeEvents/v2/PST\_com\_abc\_core\_\_engine?\$filter=Path eq 'engine\_temp' and minute(At\_DateTime) eq 15
- /CompositeEvents/v2/PST\_com\_abc\_core\_\_engine?\$filter=Path eq 'engine\_temp' and second(At\_DateTime) eq 40
- /CompositeEvents/v2/PST\_com\_abc\_core\_\_engine?\$filter=Path eq 'eventData' and BusinessTimestamp eq datetimeoffset'2015-08-02T09:00:00Z'
- /CompositeEvents/v2/PST\_com\_abc\_core\_\_engine?\$filter=Path eq 'eventData' and BusinessTimestamp gt datetimeoffset'2015-08-02T09:00:00Z' and BusinessTimestamp lt datetimeoffset'2015-10-02T09:00:00Z'

- `/CompositeEvents/v2/PST_com_abc_core__engine?$filter=Path eq 'eventData' and BusinessTimestamp eq null`

## Examples

The following are a few examples of valid filter conditions:

Operand	Example
<i>Null</i>	<code>/Things/PropertySet/&lt;property set type name&gt;?\$filter=name eq null</code>
<i>String</i>	<code>/Things/PropertySet/&lt;property set type name&gt;?\$filter=name eq 'Tenant'</code>
<i>ThingId or BPID</i>	<code>/Things/PropertySet/&lt;property set type name&gt;?\$filter=_thingId eq '327036AE27D94E9B9B85FC91A3ED0BC0'</code>
<i>Numeric</i> <ul style="list-style-type: none"> <li>• <i>Decimal</i></li> </ul>	<code>Things/PropertySet/&lt;property set type name&gt;?\$filter=myNumericDec eq 9.01</code>
<i>Numeric</i> <ul style="list-style-type: none"> <li>• <i>Integer</i></li> </ul>	<code>Things/PropertySet/&lt;property set type name&gt;?\$filter=myNumericDec eq 9</code>
<i>NumericFlexible</i>	<code>Things/PropertySet/&lt;property set type name&gt;?\$filter=myNumericFlex eq 9</code>
<i>Timestamp or DateTime</i>	<code>/Events?\$filter=_businessTimeStamp lt 2016-12-31T00:00:00.000Z</code>
<i>Date</i>	<code>Things/PropertySet/&lt;property set type name&gt;?\$filter=myDate lt 2016-12-31</code>
<i>Boolean</i>	<code>Things/PropertySet/&lt;property set type name&gt;?\$filter=myBoolean eq true</code>

## 25 Retrieving List of Records

There are API endpoints available to retrieve a list of records. Some of the API endpoints, by default retrieve 100 or 500 records per response. If there are more records available, a **\_\_next** link is available at the end of the response payload to navigate to the next set of records. The following is a snippet of response payload in [JSON](#) format with **\_\_next** link.

```
{
  "d": {
    "results": [
      {
        ...
      },
      ...
      {
        ...
      }
    ],
    "__next": "https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com/ThingGroup/v1/ThingGroup?$skiptoken=20"
  }
}
```

The following is a snippet of response payload in [XML](#) format with **\_\_next** link.

```
<entry>
  ...
</entry>

<entry>
  ...
</entry>
<link href="https://apiot-thing-group-ci.cfapps.sap.hana.ondemand.com/ThingGroup/v1/ThingGroup?$skiptoken=20" rel="next"></link>
```

However, with REST services, the response payload contains **\_\_next** and **\_\_prev** link. The following is a snippet of response payload in JSON format with **\_\_next** and **\_\_prev** links.

```
{
  "value": [
    {
      ...
    },
    ...
    {
      ...
    }
  ],
  "next": "https://apiot-mds-ci.cfapps.sap.hana.ondemand.com/Things?$skip=1000",
  "prev": "https://apiot-mds-ci.cfapps.sap.hana.ondemand.com/Things?$skip=0"
}
```

To retrieve a list of records, we recommend you avoid using `$count` with every service request. Instead, use `$top` and `$skip` in the service request, thereby reducing the server load.



For example, assume you have configured 1000 thing types. You want to retrieve thing types using the application `config-thing-sap`. This application, by default retrieves 500 records per response and displays the **\_\_next** link in the response payload.

- If you try retrieving 750 thing types using the URL `/ThingConfiguration/v1/ThingTypes?$top=750`, the service retrieves first 500 records along with the **\_\_next** link in the response payload. You can use the **\_\_next** link to retrieve the next 250 records and repeat calling the service request until all records are retrieved, if there are any.
- If you try retrieving 10 thing types using the URL `/ThingConfiguration/v1/ThingTypes?$top=10`, the service retries 10 thing types and the **\_\_next** link will not be displayed in the response payload.

# 26 Draft Mode for Service Requests

A request parameter to avoid data being written to the database

## Overview

Many of the SAP Leonardo IoT services support a special request parameter that you can use to ensure that during the request, no data is actually written to, or modified in, the database. This is helpful when you want to call a service which by definition can modify the database (HTTP types POST, PUT, and DELETE), but for certain reasons you want to avoid that anything is actually written to the database. Here are some examples of test scenarios where this can be useful or even required:

- You might want to create a temporary BPRelationship object instance without defining the IDs of Persons or Companies that are related via the object. Providing the corresponding object IDs is normally mandatory, and trying to save a BPRelationship object without these values would lead to an error message. Calling a service with the draft parameter provided helps you to avoid such error messages.
- You have to deal with a use case where several different services are involved, which are mutually dependent so that the entire use case can only be solved if certain values are present in the database. With the draft mode, you can first check the presence of the required data and, in case they are missing, cancel the processing in a controlled manner.

## Availability

Services With Draft Mode Support

Area	Services
Business Partner	All services
Authorization	All services
Location	All services

## Details

Parameter: `draft=true`

## Request Example

```
/Persons ('adbb2c0fdb0a4b9e99327a52c41fc720')?draft=true
```

This request simulates creating or updating a person with the specified ID without actually creating or updating that person. Also, if a Person object with that ID already exists in the database, the draft parameter protects you from creating any inconsistencies with that call.

## 27 Reserved Keywords for Configuration and Thing Services

There are reserved keywords that must not be used while defining the configuration or thing content.

Following are the list of reserved keywords for use within the Configuration and Thing Services. These keywords, in any textual case, are considered as reserved keywords. For example ATTRIBUTETYPE, attributetype, and AttributeType are all reserved keywords.

_externalId	_objectGroup	_thingType
_name	_QC	_time
		_SAPIOT_Monitoring_Thing
<b>a</b>		
addresspresentation	annotation	authObjectGroup
addressrepresentationisocode	attributetype	bpaddressrepresentation
<b>b</b>		
bpcountry	bpmobilephonenumber	bporganizationname4
bpcountrydescription	bpname	businesspartnercategory
bpfamilyname	bporganizationname1	businesspartnerid
bpgivenname	bporganizationname2	businesstimestamp
bplandlinephonenumber	bporganizationname3	
<b>c</b>		
children	comment	countrydescription
cityname	correlationid	curr_version
code	country	customer

---

**d**

datatype	dependencies	descriptionpt
datacategory_referencepst	description	descriptionshort
dataCategory	description_referencepst	descriptiontt
dataService	description_referringpst	descriptionttpst
default_id	descriptionlanguage	distance
delete	descriptionpst	district
deletable		

---

**e**

enddatetime	etlanguage	eventseverity2count
entity_used	etptdescription	eventseverity3count
entity_used_by	etptlanguage	eventstatuscode
entity_used_by_type	etsdescription	eventtype
entity_used_id	etslanguage	eventtypecode
entity_used_type	etsvdescription	eventtypepropertyset
etag	etsvlanguage	eventtypeseverity
etagversion	event	eventtypestatus
etcdescription	eventexternalid	extendedThingCategory
etclanguage	eventseverity	extendedThingType
etdescription	eventseverity1count	externalid

---

**f**

---

familyname

---

**g**

---

givenname

---

## h

hasdecimals	hierarchyetagversion	hierarchynodeid
haslength	hierarchyexternalid	hierarchytenant
hierarchydescisocode	hierarchyid	hierarchytype
hierarchydescription	hierarchyname	houzenumber
hierarchydescriptionlanguage	hierarchynodesdescription	

## i

id	isocode	isocodett
is_cycle	isocode_referencepst	isocodettpst
is_leaf	isocodepst	isodescriptionpt
iscompleted	isocodept	

## l

landlinephonenumber	latitude	level_name
language	length	location
languagecode	level	longitude

## m

maxdecimals
maxlength
migrationid
mobilephonenumber

## n

name	nodesdescriptionlanguage	nodessemantic
nationality	nodeid	
nodesdescription	nodesdescisocode	

**o**

objectgroup	objectType_1	organizationname1
objectgroup_1	openevents	organizationname2
objectid	openeventscount	organizationname3
objectType	ordinal	organizationname4

**p**

package	postalcodelength	propertySetTypes
parentobjectid	pred_node	propertysettypepackage
parentobjecttype	propdescription	propertyTypes
parents	property	proplanguage
parentthing	propertydatatype	pst
parentthingdescription	propertydatatypepelength	pstdescription
parentthingname	propertyhasqualitycode	pstlanguage
path	propertyname	pstversion
position	propertyset_id	pt_id
postalcode	propertySetType	

**q**

qualityCode
query_node
query_node_name

**r**

read	referringproperty	resolution
readable	referringpst	result_node
referencepropertytype	region	result_node_name
referredproperty	regiondescription	retentionperiod
referredpst	regionlanguage	rootthing
		runid

## s

schema	sourceId	systemadministratedata_creationdate
scope	sourceversion	
sequence	startdatetime	systemadministratedata_lastchange- datetime
servicename	status	systemadministratedata_lastchan- gedbyuser_id
services	streetname	
severity	systemadministratedata_created- byuser_id	systemadministratedata_lastchan- gedbyuser_tenant
source	systemadministratedata_created- byuser_tenant	systemstatus

## t

tablename	thingdescription	thingregiondescription
targetversion	thingdistrict	thingstreetname
tenant	thingexternalid	thingTypes
thing	thinghouseNumber	thingtypepackage
thingaddressrepresentation	thingid	thingvalidityenddatetime
thingaddressrepresentationisocode	thinglocationid	thingvaliditystartdatetime
thingCategories	thingmodel_used_by_package	thingwrite
thingcityname	thingmodel_used_package	time
thingcountry	thingname	timeDependency
thingcountrydescription	thingpostalcode	timedepresolution
thingcustomerid	thingread	timestamp
thingdelete	thingregion	type

## u

user		
userGroup		

**v**

---

validityenddatetime	value_datatype	valuedatatype
validitystartdatetime	value_hasqualitycode	value_unitofmeasure
value	value_length	version
		viewname

---

**w**

---

writable

write

---



## 28 Value Data Types for Property Types and Properties

Value Data Type	Data Type	Length Mandatory	Value
String	String	No	
	<p><b>i Note</b></p> <p>Maximum permissible length is 254 characters.</p> <p>You can increase the length for a property type even if the data already exists for this property. However, you cannot decrease the length for a property regardless of whether the data exists or not.</p> <p>Special characters forward slash (/) and caret (^) are not supported.</p>		<p><b>i Note</b></p> <p>Default length is 127 characters.</p>
LargeString	LargeString	No	Maximum permissible limit is 512k.
Numeric	Integer	No	
	<p><b>i Note</b></p> <p>Maximum permissible value is 2,147,483,647.</p> <p>Minimum permissible value is -2,147,483,648</p>		

Value Data Type	Data Type	Length Mandatory	Value
Numeric	Decimal	Yes	
	<div> <div>i Note</div> <p>When you set a decimal value for a property type, you must indicate the precision upto which you require the value. This value is defined in the format (x,y), where:</p> <ul style="list-style-type: none"> <li>• X indicates the overall digits in the value. The permissible range is between 1 to 10 digits.</li> <li>• Y indicates the number of decimal or fractional digits in the value. The permissible range is between 1 to X.</li> </ul> <p>For example, for a value 32.45, the format defined is (4,2).</p> <p>For a value 12345.123, the format defined is (8,3).</p> </div>		
NumericFlexible	Float	No	Value range is -3.4E+38 to +3.4E+38
	<div> <div>i Note</div> <p>This is used to store 32 bit floating point values.</p> </div>		
ThingID	String	No	Default is 32 characters
	This is used to store 32 character ThingID.		
BPID	String	No	Default is 32 characters
Timestamp	Timestamp	No	Value range is 0001-01-01 00:00:00 to 9999-12-31 23:59:59.9999999
	Corresponds to UTC date and time (with a precision of 0.1 microseconds).		
DateTime	Date time	No	Values range is 0001-01-01 00:00:00 to 9999-12-31 23:59:59
	Corresponds to UTC date and time (with seconds precision)		
Date	Date	No	Value range is 0001-01-01 to 9999-12-31

Value Data Type	Data Type	Length Mandatory	Value
JSON	LargeString	No	Maximum permissible limit is 64k.
Boolean	Boolean	No	Values supported are <b>true</b> and <b>false</b> .
ByteArray	ByteArray	No	Default length is 300 bytes.

#### i Note

You cannot change a Boolean data type to another data type and vice versa.

#### i Note

Maximum permissible length is 1024 bytes.

Minimum permissible value is 0.

Maximum permissible value is 255.

Value Data Type	Data Type	Length Mandatory	Value
GeoJSON	JSON	No	<p>You can only store latitude, longitude, and geometry type values.</p> <p>The first value in the coordinates represents the latitude and the second value represents the longitude.</p>

#### **i** Note

- The only supported geometry type is Point.
- For a property set type of data category *TimeSeriesData* and *DerivedData*, you can define property types of data type *GeoJSON*. However, you can create time series data or derived data for a property of type *GeoJSON* only via data ingestion from SAP Cloud Platform Internet of Things for the Cloud Foundry Environment (Internet of

Value Data Type	Data Type	Length	Mandatory	Value
				Things Service).

## Examples for Property Definition

Value Data Type	Payload Type	Example
Numeric: Integer	Configuration	<pre>"Properties": [{   "Name": "Pressure",   "Type": "Numeric" }]</pre>
	Thing	<pre>{   "value": [{     "Pressure": 100,     "time": "2015-08-01T09:00:00.000Z"   }] }</pre>
Numeric: Decimal	Configuration	<pre>"Properties": [{   "Name": "Temperature",   "Type": "Numeric",   "PropertyLength": "4,2" }]</pre>
	Thing	<pre>{   "value": [     {       "Temperature": 99.99,       "time": "2015-08-01T09:00:00.000Z"     }   ] }</pre>
NumericFlexible	Configuration	<pre>"Properties": [{   "Name": "MachineReading",   "Type": "NumericFlexible" }]</pre>
	Thing	<pre>"value": [   {     "MachineReading": -3.4E+38,     "time": "2015-08-01T09:00:00.000Z"   } ]</pre>

Value Data Type	Payload Type	Example
Boolean	Configuration	<pre>"Properties": [{   "Name": "MachineRunningStatus",   "Type": "Boolean" }]</pre>
	Thing	<pre>{   "value": [{     "MachineRunningStatus": true,     "_time": "2015-08-01T09:00:00.000Z"   }] }</pre>
String (without a specified length)	Configuration	<pre>"Properties": [{   "Name": "MachingID",   "Type": "String" }]</pre>
	Thing	<pre>{   "value": [     {       "MachineID": "6493474340414429B89B844B0758ACC5",       "_time": "2015-08-01T09:00:00.000Z"     }   ] }</pre>
String (with a specified length)	Configuration	<pre>"Properties": [{   "Name": "MachingID",   "Type": "String",   "PropertyLength": 60 }]</pre>
	Thing	<pre>{   "value": [     {       "MachineID": "6493474340414429B89B844B0758ACC5",       "_time": "2015-08-01T09:00:00.000Z"     }   ] }</pre>
ThingID	Configuration	<pre>"Properties": [{   "Name": "MachingID",   "Type": "ThingID" }]</pre>

Value Data Type	Payload Type	Example
	Thing	<pre>{   "value": [     {       "MachineID": "6493474340414429B89B844B0758ACC5",       "_time": "2015-08-01T09:00:00.000Z"     }   ] }</pre>
BPID	Configuration	<pre>"Properties": [{   "Name": "ContanctPersonID",   "Type": "BPID" }]</pre>
	Thing	<pre>{   "value": [     {       "ContactPersonId":         "6493474340414429B89B844B0758ACC5",       "_time": "2015-08-01T09:00:00.000Z"     }   ] }</pre>
Timestamp	Configuration	<pre>"Properties": [{   "Name": "timeReceived",   "Type": "timestamp" }]</pre>
	Thing	<pre>{   "value": [     {       "timeReceived": "2015-08-01T09:00:00.000Z",       "_time": "2016-01-01T09:00:00.000Z"     }   ] }</pre>
DateTime	Configuration	<pre>"Properties": [{   "Name": "timeReceived",   "Type": "DateTime" }]</pre>
	Thing	<pre>{   "value": [     {       "timeReceived": "2015-08-01T09:00:00Z",       "_time": "2016-01-01T09:00:00.000Z"     }   ] }</pre>





Value Data Type	Payload Type	Example
GeoJSON	Configuration	<pre>"Properties": [{   "Name": "geolocation",   "Type": "GeoJSON" }]</pre>
	Thing	<pre>"geolocation": {   "type": "Point",   "coordinates": [     78.47444,     17.37528   ] }</pre>

## 29 Value Helps

Describes the value help services available for SAP Leonardo IoT.

For SAP Leonardo IoT, a number of generic services are available that provide lists of predefined values that can be used for various purposes within the platform, especially for the sake of providing value helps from which a user can pick the desired value. The following value help services are currently available:

- Countries
- Regions
- Languages
- Units of Measure
- Academic Titles
- Forms of Address

The following applies for all value help services:

- Each value entry is organized as a pair of an internal technical key and a descriptive value name in natural language. For the language, the system uses the following approach:
  1. Only languages that are explicitly mentioned in the relevant whitelist for the Cloud Foundry environment can be displayed, regardless of all other preconditions.
  2. If a value help service is called by an application running in a web browser, the system presents the value names in the language that has been defined in the browser settings.
  3. You can explicitly specify the desired language by adding an `Accept-Language` header parameter to the service call.
  4. If the browser language is not available for a value help, or if a value help service is called by a backend application (that is, no web browser involved), the system falls back to English as the default language.

The four ways of language determination are given with descending priority. That is, the system first checks the whitelist, then tries to deliver the required data according to the browser settings, then tries to fulfill the `Accept-Language` header, and finally (if all fails) falls back to English.

### ❖ Example

Let's assume the value help data in the system has been shipped in three languages, English, German, and French. When you call the value help service for languages, the language names are presented as follows, depending on the language determination conditions outlined above:

Language Value Help in Different Languages

ISO language code	Determined Display Language		
	English	German	French
en-EN	English	Englisch	Anglais
de-DE	German	Deutsch	Allemand

Determined Display Language			
ISO language code	English	German	French
fr-FR	French	Französisch	Français

### i Note

The language determination logic as outlined above as well as the given example can of course only work as described under the prerequisite that the necessary string resources are all present in the system. SAP starts the first generally available deliveries of SAP Leonardo IoT with string resources in English only, followed by string resources in German, which have been released in late 2016. Further languages may follow in the future.

- The values contained in all value help services are offered in read-only mode. There is no way of modifying single values or extending the list of values.
- From a technical point of view, all value help services consist of a single HTTP GET method and return a collection of values. Picking a single value from the lists is only possible by applying a filter that restricts the service response to a single list entry.

### i Note

There is one exception to this rule: In contrast to all other value help services, you request a single value with the UnitsOfMeasure service, for example, `/UnitsOfMeasure('KG')`.

If the desired unit of measure contains special characters that are not allowed in a URL, these characters must be masked. For example, to retrieve the percentage sign (%) from the database, you need to use the following syntax: `/UnitsOfMeasure('%25')`. Units of measure that contain a dash sign (for example, "m/s" for speed), need to be given with a double mask like this: `/UnitsOfMeasure('m%252Fs')`. Here, "%25" refers to the hexadecimal code of the percentage sign needed in the URL to indicate that another hexadecimal code is following, while "2F" is the hexadecimal code of the dash sign (ASCII code 47).

- All value help services are available throughout the entire system. That is, there is no tenant dependency for accessing these services.
- The value help services are called without parameters except for sorting and filtering. The only exception is the Regions service, which requires a country code as a parameter in order to restrict the list of regions returned to that country.
- None of the value help services does take any payload data in the HTTP request.

#### Base URI:

- Formal description: `http://<server address>[:<port number>][<path>]/<value help service>`
- Example for a base URI in a cloud foundry environment: `http://business-partner.cfapps.neo.ondemand.com/<value help service>`

**Permissions:** `<bp>.r`, `<bp>.vh.r`

### i Note

Since the `<bp>.r` scope is the absolute minimum prerequisite for any user who is supposed to do productive work in the SAP Leonardo IoT framework, this permission may be considered as always granted (but still must be explicitly assigned to a user).

## Methods

HTTP Method	URI
<i>GET</i>	<code>/Countries</code>
<i>GET</i>	<code>/Regions('&lt;countryCode&gt;')</code>
<i>GET</i>	<code>/Languages</code>
<i>GET</i>	<code>/UnitsOfMeasure</code> <code>/UnitsOfMeasure('&lt;UoM&gt;')</code>
<i>GET</i>	<code>/AcademicTitles</code>
<i>GET</i>	<code>/FormOfAddresses</code>

### i Note

The formulation of the service names follows the common practice for collective requests by putting a plural-s at the end of the service name. Here, the UnitsOfMeasure service is an exception to this rule as the service name contains the plural-s at the linguistically correct position within this composite noun (that is, the plural-s follows the term "Unit" rather than the term "Measure").

Also, the UnitsOfMeasure service stands out of the other value helps with respect to the following feature: As you can see in the table above, the UnitsOfMeasure service can be requested with two different calling conventions:

- Collective GET request: `/UnitsOfMeasure`
- Single GET request: `/UnitsOfMeasure('<UoM>')`

While the collective GET request works the same as all the other value help services, the single GET request for a particular unit of measure is in place to return the full record of data that is kept in the database for a particular unit of measure. These additional data comprises, for example, the number of decimals used for rounding, the dimension to which a unit of measure belongs (e.g., weight, length, speed), conversion factors, ISO code.

## Request

URI: `/<value help service>`

HTTP Method: *GET*

### Permissions:

<bp>.r (Country, Region, Language, Academic Title, Form Of Address)

<bp>.vh.r (only Units of Measure)

### Query String Parameters

Name	Required	Data Type	Description
\$skip	No	integer	Number of the first n records to be excluded from the result set
\$top	No	integer	Number of records to include in the result set
\$orderby	No	string	Field name to be used for sorting the result set in ascending or descending order; for valid fields of the different value help services, see the table below.
\$filter	No	string	Filter condition to be applied; for valid fields of the different value help services, see the table below.

Valid Fields for Sorting and Filtering Value Help Result Lists

Value Help	Valid Fields
Country	isoCode
Region	countryIsoCode regionCode
Language	isoCode
Unit of Measure	unitOfMeasure noDecimalForRounding unitOfMeasureDimension numeratorForConversion denominatorForConversion baseTenExponentForConversion numberOfDecimalPlaces unitOfMeasureISOCODE isPrimaryUnitForISOCODEConversion additiveConstantForConversion baseTenExponentFloatDsip
Academic Title	academicTitleCode
Form of Address	formOfAddressCode

### Request Example

```
/Regions('DE')
```

Retrieves all regions that are defined for the Federal Republic of Germany.

```
/Languages?$top=25&$orderby=isoCode
```

Retrieves the first 25 languages, sorted by the language ISO code.

## Response

### Response Status and Error Codes

Code	Description
200	Value list retrieved successfully

## Related Information

[ISO Language Codes \[page 1242\]](#)

[Country Codes \[page 1244\]](#)

## 29.1 ISO Language Codes

List of supported language codes for SAP Leonardo IoT

In the following table, you find the list of supported languages for SAP Leonardo IoT and the corresponding ISO codes that you need to tell the system which language shall be used in a particular service request.

### Note

Note that this list of languages is only meant as an overview of the language codes that SAP Leonardo IoT can handle. It does of course **not** mean that all the texts or messages that may occur in the context of SAP Leonardo IoT are already available in these languages, be it at design time or at runtime of the applications that you build on the basis of the services.

ISO Language Codes

Description	ISO Code
Afrikaans	af
Arabic	ar
Bulgarian	bg

Description	ISO Code
Catalan	ca
Chinese	zh
Croatian	hr
Czech	cs
Danish	da
Dutch	nl
English	en
Estonian	et
Finnish	fi
French	fr
German	de
Greek	el
Hebrew	he
Hindi	hi
Hungarian	hu
Icelandic	is
Indonesian	id
Italian	it
Japanese	ja
Korean	ko
Latvian	lv
Lithuanian	lt
Malaysian	ms
Norwegian	no
Polish	pl
Portuguese	pt

Description	ISO Code
Romanian	ro
Russian	ru
Serbian	sr
Slovakian	sk
Slovenian	sl
Spanish	es
Swedish	sv
Thai	th
Turkish	tr
Ukrainian	uk
Vietnamese	vi

## Related Information

[Value Helps \[page 1238\]](#)

## 29.2 Country Codes

List of supported country codes for Leonardo IoT

In the following table, you find the list of supported countries for Leonardo IoT and the corresponding country codes (according to ISO 3166-1) that you need, for example, to provide the location of a particular business partner.

### Countries and Regions

Most countries of the world have defined regions to further specify a particular part of the country. It depends on the political and administrative structure of each country to specify what exactly a region for a given country refers to. For example, in country DE (Federal Republic of Germany), the regions reflect the 16 federal states like Bavaria or Lower Saxony. The country codes returned by this value help service also serve as input parameters for the Regions service, which returns the set of regions that exist in a particular country.

#### i Note

The country codes as defined in this table are also needed for specifying the country-specific dialing prefix in the payload for communication data of persons and organizations.



## Country Codes

Country/Region	Country Code
Afghanistan	AF
Åland Islands	AX
Albania	AL
Algeria	DZ
America	AS
American Minor Outlying Islands	UM
American Virgin Islands	VI
Andorra	AD
Angola	AO
Anguilla	AI
Antarctica	AQ
Antigua and Barbuda	AG
Argentina	AR
Armenia	AM
Aruba	AW
Australia	AU
Austria	AT
Azerbaijan	AZ
Bahamas	BS
Bahrain	BH
Bangladesh	BD
Barbados	BB
Belarus	BY
Belgium	BE
Belize	BZ
Benin	BJ

Country/Region	Country Code
Bermuda	BM
Bhutan	BT
Bolivia	BO
Bosnia and Herzegovina	BA
Botswana	BW
Bouvet Islands	BV
Brazil	BR
British Indian Ocean Territory	IO
British Virgin Islands	VG
Brunei Darussalam	BN
Bulgaria	BG
Burkina Faso	BF
Burma	MM
Burundi	BI
Cambodia	KH
Cameroon	CM
Canada	CA
Cape Verde	CV
Cayman Islands	KY
Central African Republic	CF
Chad	TD
Chile	CL
China	CN
Christmas Island	CX
Coconut Islands	CC
Colombia	CO

Country/Region	Country Code
Comoros	KM
Cook Islands	CK
Costa Rica	CR
Côte d'Ivoire	CI
Croatia	HR
Cuba	CU
Curaçao	CW
Cyprus	CY
Czech Republic	CZ
Democratic Republic of the Congo	CD
Denmark	DK
Djibouti	DJ
Dominica	DM
Dominican Republic	DO
East Timor	TL
East Timor (deprecated)	TP
Ecuador	EC
Egypt	EG
El Salvador	SV
Equatorial Guinea	GQ
Eritrea	ER
Estonia	EE
Ethiopia	ET
Falkland Islands	FK
Faroe Islands	FO
Fiji	FJ

Country/Region	Country Code
Finland	FI
France	FR
French Guyana	GF
French Polynesia	PF
French Southern and Antarctic Lands	TF
Gabon	GA
Gambia	GM
Georgia	GE
Germany	DE
Ghana	GH
Gibraltar	GI
Greece	GR
Greenland	GL
Grenada	GD
Guadeloupe	GP
Guam	GU
Guatemala	GT
Guernsey (Channel Islands)	GG
Guinea	GN
Guinea-Bissau	GW
Guyana	GY
Haiti	HT
Heard and McDonald Islands	HM
Honduras	HN
Hong Kong, China	HK
Hungary	HU

Country/Region	Country Code
Iceland	IS
India	IN
Indonesia	ID
Iran	IR
Iraq	IQ
Ireland	IE
Isle of Man	IM
Israel	IL
Italy	IT
Jamaica	JM
Japan	JP
Jersey	JE
Jordan	JO
Kazakhstan	KZ
Kenya	KE
Kiribati	KI
Kuwait	KW
Kyrgyzstan	KG
Laos	LA
Latvia	LV
Lebanon	LB
Lesotho	LS
Liberia	LR
Libya	LY
Liechtenstein	LI
Lithuania	LT

Country/Region	Country Code
Luxembourg	LU
Macau, China	MO
Macedonia	MK
Madagascar	MG
Malawi	MW
Malaysia	MY
Maldives	MV
Mali	ML
Malta	MT
Marshall Islands	MH
Martinique	MQ
Mauritania	MR
Mauritius	MU
Mayotte	YT
Mexico	MX
Micronesia	FM
Moldova	MD
Monaco	MC
Mongolia	MN
Montenegro	ME
Montserrat	MS
Morocco	MA
Mozambique	MZ
Namibia	NA
Nauru	NR
Nepal	NP

Country/Region	Country Code
Netherlands	NL
New Caledonia	NC
New Zealand	NZ
Nicaragua	NI
Niger	NE
Nigeria	NG
Niue	NU
Norfolk Islands	NF
North Korea	KP
North Mariana Islands	MP
Norway	NO
Oman	OM
Pakistan	PK
Palau	PW
Palestine	PS
Panama	PA
Papua New Guinea	PG
Paraguay	PY
Peru	PE
Philippines	PH
Pitcairn Islands	PN
Poland	PL
Portugal	PT
Puerto Rico	PR
Qatar	QA
Republic of the Congo	CG

Country/Region	Country Code
Reunion	RE
Romania	RO
Russian Federation	RU
Rwanda	RW
Saba	BQ
Saint Helena	SH
Saint Kitts and Nevis	KN
Samoa	WS
San Marino	SM
Sao Tome and Principe	ST
Saudi Arabia	SA
Senegal	SN
Serbia	RS
Serbia and Montenegro	CS
Seychelles	SC
Sierra Leone	SL
Singapore	SG
Sint Maarten (Dutch part)	SX
Slovakia	SK
Slovenia	SI
Solomon Islands	SB
Somalia	SO
South Africa	ZA
South Georgia and the Southern Sandwich Islands	GS
South Korea	KR
South Sudan	SS



Country/Region	Country Code
Spain	ES
Sri Lanka	LK
St. Lucia	LC
Saint Martin (French part)	MF
St. Pierre and Miquelon	PM
St. Vincent and the Grenadines	VC
Sudan	SD
Suriname	SR
Svalbard	SJ
Swaziland	SZ
Sweden	SE
Switzerland	CH
Syria	SY
Taiwan, China	TW
Tajikistan	TJ
Tanzania	TZ
Thailand	TH
Togo	TG
Tokelau Islands	TK
Tonga	TO
Trinidad and Tobago	TT
Tunisia	TN
Turkey	TR
Turkmenistan	TM
Turks and Caicos Islands	TC
Tuvalu	TV

Country/Region	Country Code
Uganda	UG
Ukraine	UA
United Arab Emirates	AE
United Kingdom	GB
Uruguay	UY
USA	US
Uzbekistan	UZ
Vanuatu	VU
Vatican City	VA
Venezuela	VE
Vietnam	VN
Wallis and Futuna Islands	WF
West Sahara	EH
Yemen	YE
Zambia	ZM
Zimbabwe	ZW

## Related Information

[Value Helps \[page 1238\]](#)

## 30 HTTP Status Codes

List of status codes for Leonardo IoT

### Status Codes

In the following table, you find the list of HTTP status codes that are used by Leonardo IoT:

HTTP Status Codes

Post	Put	Delete	Get (single value)	Get (collection)	Description
201	200	204	200	200	Success
n/a	404	404	404	n/a	Not found
n/a	409	409	n/a	n/a	Conflict (If-Match condition failed)
403	403	403	403	403	Forbidden (insufficient authorization)
400	400	400	400	400	Bad request (for example: header field missing, wrong filter, validation failed)
Various error codes					Not allowed
500	500	500	500	500	Internal server error

The different response codes used by Leonardo IoT can be grouped into three categories that differ with respect to the question of whether a user action is required or not:

Response Code Categories

Number Range	Category	User Action
200..299	Success	No action required
400..499	Client-side error	Error can be solved by ensuring valid and correct call parameters.

Number Range	Category	User Action
500..599	Server-side error	Error occurred on server backend system. Call the system administrator for support.

## Tracking Errors by Correlation ID

Whenever an error occurs during the execution of an application that is based on the API services described in this reference, you will notice that by either seeing one of the error codes listed above in your browser, or by finding that error code in the error logs that have been recorded for your application. The problem here, however, is that the activity that actually triggered the error message may only accidentally be responsible for the error to occur. It is much more likely that this final step only unveiled a problem for which the root cause may be hidden in any of the numerous software layers or instances, or be limited to a certain physical machine in your system landscape on which the code was executed. Because of the complexity of common system landscapes it must be counted as the rule rather than an exception that the root cause of an error is typically hard to trace back.

To overcome the problem of tracing back well-hidden root causes for the problems that may appear on the surface, SAP has implemented a tracking mechanism that gives service departments a chance to find their way from the surface down to the ground from where an error originated. This is accomplished by attaching an individual ID to each service request. The system architecture can then make sure that the attached ID is handed over from one service to another. As a result, all service requests that are involved in the execution of a particular task can be identified by their ID (that is, all services in the call stack that are involved in that task carry the same ID). With that, an administrator can easily identify the relevant requests and filter out all the noise that stems from other, unrelated system activities.

### Solution

To overcome the problem of identifying the root cause for a problem that has occurred during execution of your application, proceed as follows:

1. We recommend that in your project, you have included a library that enables you to record the correlation ID in your error logs. Here are some examples of open-source libraries that you might want to use:
  - Java applications: <https://github.com/SAP/cf-java-logging-support>
  - NodeJS applications: <https://github.com/SAP/cf-nodejs-logging-support>
2. Whenever you call a service in your application, make sure that the following HTTP header parameter is passed along with the request: `X-CorrelationID`
3. Before actually sending the request, assign a unique value to the `X-CorrelationID` parameter (for example a combination of a timestamp and a random number).



As a result, the `X-CorrelationID` with the assigned individual value is automatically passed on to all subsequent service requests so that an administrator can easily isolate the complete request chain for further investigations.

# Important Disclaimers and Legal Information

## Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.

About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering a SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

## Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

## Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

## Gender-Related Language

We try not to use gender-specific word forms and formulations. As appropriate for context and readability, SAP may use masculine word forms to refer to all genders.

## Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

© 2019 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see <https://www.sap.com/about/legal/trademark.html> for additional trademark information and notices.