



The Retail Innovators

SAP Dynamic Pricing by GK

Configuration Guide

Version: v4.0.0



COPYRIGHT

© 2022 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.

1. You may not use the SAP Material for a purpose competitive with SAP or its products unless otherwise clearly permitted by applicable law.
2. You may not use the SAP corporate logo.
3. No use of other SAP trademarks is granted under this section. For information regarding use of SAP trademarks, see <http://www.sap.com/corporate/en/legal/trademark.html>.

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.

Internal Document Information: 1223593582 | 2022-03-02

TABLE OF CONTENTS

1	Overview	6
2	Interfaces	6
2.1	Import/Export Configuration: Credential Encryption	6
2.1.1	Summary	6
2.1.2	Password encryption	6
2.1.3	Supported Tags	7
2.2	Import/Export Configuration: Business Unit Parameters	7
2.2.1	Summary	7
2.2.2	Configuration	7
2.2.2.1	Example	8
2.2.2.2	Inheritance and Overriding	8
2.3	Import/Export Configuration: SAP Passport	9
2.3.1	Summary	9
2.3.2	Configuration	9
2.3.2.1	Parameters	9
2.3.2.2	Example	10
2.4	Export	10
2.4.1	Export Protocol	10
2.4.1.1	Summary	10
2.4.1.2	Protocols	10
2.4.1.3	Configuration	11
2.4.2	Export Communication Interface	11
2.4.2.1	Summary	11
2.4.2.2	Configuration	12
2.4.2.3	Download Export Data via REST Interface (PULL)	15
2.4.3	Export CSV Writing	15
2.4.3.1	Summary	15
2.4.3.2	Configuration of exporting CSV Data	15
2.4.4	Export Timezones	16
2.4.4.1	Summary	16
2.4.4.2	Timezone in the Request	16
2.4.4.3	Timezone in the XML configuration	16
2.4.4.4	Timezone in the Business Unit	16
2.4.5	Export Pipelines	17
2.4.5.1	Summary	17
2.4.5.2	Export optimized prices of a business unit hierarchy for each business unit to one/variable endpoint or multiple files	17
2.4.5.3	Export optimized prices of a business unit hierarchy to one endpoint or file	19
2.4.5.4	Download Export Data via REST Interface (PULL)	21

2.4.5.5	Additional Configuration Options using Business Unit Parameters	22
2.4.6	Prices Export	22
2.4.6.1	Prices Export - CSV	22
2.4.6.2	Prices Export - SAP Enterprise Service	25
2.5	Import	28
2.5.1	Import Protocol	28
2.5.1.1	Summary	28
2.5.1.2	Protocols	28
2.5.1.3	Configuration	29
2.5.2	Import Communication Interface	30
2.5.2.1	Summary	30
2.5.2.2	Configuration	30
2.5.3	Import CSV Reading	36
2.5.3.1	Summary	36
2.5.3.2	Configuration of importing CSV Data	36
2.5.4	Import XML Reading	37
2.5.4.1	Summary	37
2.5.4.2	Configuration of importing XML Data	37
2.5.5	Data Mapping by Configuration	37
2.5.5.1	Mapping by column name	38
2.5.5.2	Mapping with a fixed value	38
2.5.5.3	Mark a field as mandatory	38
2.5.6	Import Timezones	39
2.5.6.1	Summary	39
2.5.6.2	Timezone in the Request	39
2.5.6.3	Timezone in the XML configuration	39
2.5.6.4	Timezone in the Business Unit	39
2.5.7	Import Pipelines	40
2.5.7.1	Summary	40
2.5.7.2	Overview	40
2.5.7.3	Scenarios	41
2.5.7.4	Additional Configuration Options using Business Unit Parameters	49
2.5.8	Competitor Prices Import	49
2.5.8.1	Competitor Prices Import - CSV	49
2.5.9	Transaction Log Import	51
2.5.9.1	Transaction Log Import - CSV	51
2.5.9.2	Transaction Log Import - POSLog	57
2.5.9.3	Transaction Log Import - SAP CAR	63
2.5.9.4	Transaction Log RDE-Server	68
2.5.9.5	Transaction Log Import - SAP CSI	71

2.5.10	Master Data Import.....	78
2.5.10.1	Item Import - AddOn CSV.....	78
2.5.10.2	Master Data Import - CSV.....	84
2.5.10.3	Master Data Import - SAP MerchandiseERPReplicationBulkRequest.....	95
2.5.10.4	Item Attribute Import - SAP IDOC ARTMAS.....	107
2.5.10.5	Item Stock Import - SAP InventoryBulkRequest_Out.....	110
2.5.10.6	Item Import - Prices CSV.....	112
2.5.10.7	Merchandise Group Import - SAP IDOC WPDWGR01.....	117
2.5.10.8	Master Data Import - GK/Retail XML.....	119
2.5.11	Exchange Rate Import	131
2.5.11.1	Exchange Rate Import - CSV.....	131
2.5.11.2	Exchange Rate Import - SAP Enterprise Service.....	132
2.5.12	Business Unit Import	135
2.5.12.1	Summary.....	135
2.5.12.2	Communication	136
2.5.12.3	Modes of Import.....	136
2.5.12.4	Section: ControlHeader	137
2.5.12.5	Section: Data.....	139
2.5.12.6	Business Unit: Parameters	146
2.5.12.7	Example	148
2.5.13	Product Link Import.....	149
2.5.13.1	Summary.....	149
2.5.13.2	Communication	149
2.5.13.3	Top Level Structure	149
2.5.13.4	Mapping and Validation	151
2.5.13.5	Configuration	156

1 Overview

This document guides you in the configuration process of the application. The document describes the configuration of:

- Import interfaces
- Export interfaces

Please see the documents "Operation Guide" and "Installation Guide" for administrative configuration details.

2 Interfaces

2.1 Import/Export Configuration: Credential Encryption

2.1.1 Summary

There are fields in config files which will contain a password as value. Because of that you're able to encrypt your password so that you can put the encrypted password into your config file instead of the password itself.

2.1.2 Password encryption

To encrypt your password you have to go to the settings. There you can click on Password Encryption.

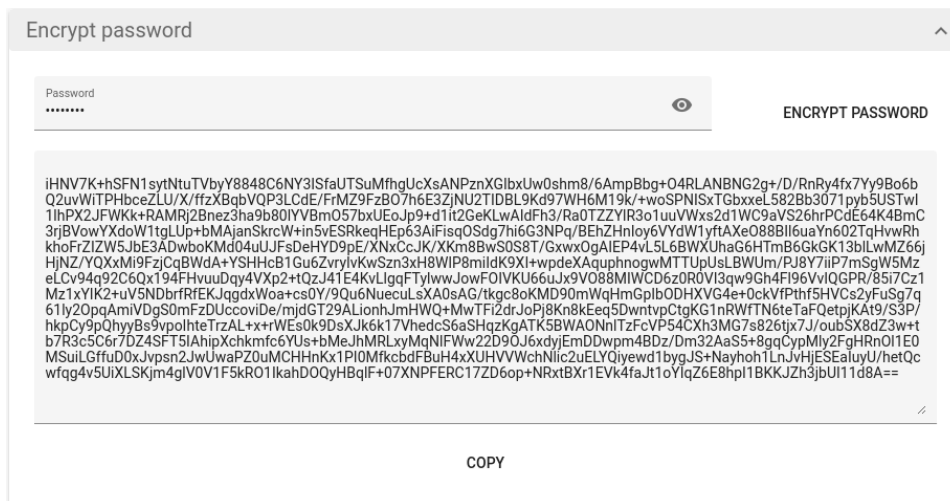
If you want to create a password you can do this by specifying the length of the password and which characters should be used. Then you can click on the button create CREATE PASSWORD which will generate a password and the key for it.

When you already have a password and you just want to create a key you need to write it into the Password field. To create the key you then will have to press the ENCRYPT PASSWORD button.



By default the password will be displayed through points. To show the password you will need to press the "eye" button on the right side of the Password field.

When you press the ENCRYPT PASSWORD button the key for your password will be created and shown. By pressing the button COPY you can copy the key to your clipboard and then paste it into an tag which supports an encrypted password.



2.1.3 Supported Tags

The following tags of the `HttpInboundAdapter` and `HttpOutboundAdapter` are supporting an encrypted value.

- `encryptedPassword`
- `clientSecret`

The following tags of the `SftpInboundAdapter` and `SftpOutboundAdapter` are supporting an encrypted value.

- `encryptedPassword`
- `encryptedPassPhrase`

2.2 Import/Export Configuration: Business Unit Parameters

2.2.1 Summary

Both import and export XML configurations support automatic parameter replacement for business unit parameters in all XML tags.

Business unit parameters are inherited by child business units by default. Business unit parameters can be overridden in child business units and then are inherited to children as well.

You are able to set business unit parameters via GUI or via business unit XML import.

Business unit parameters are an important tool for managing business unit hierarchies. For instance you might want to centrally manage the export settings for the SAP price export but use the correct sales organization based on your BU settings without losing the easy central handling of one configuration file for all business unit exporters.

2.2.2 Configuration

To replace a business unit parameter in your XML configuration file for import/export you need to address your parameter this way:

```
${BU.PARAMETERS.xxx}
```

xxx is the name of the parameter that you used in the UI or business unit XML import.

2.2.2.1 Example

Edit Business Unit ✕

GENERAL

CONFIGURATION

STATISTICS

OPTIMIZATION

PARAMETERS

Parameter *
SalesOrg

Value
RS10

–

ADD SETTING

CANCEL

APPLY

Here we set a parameter named "SalesOrg" to the value "RS10".

You can use the parameter replacement in your XML file by addressing the parameter with `${BU.PARAMETERS.SalesOrg}`:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <salesOrganisationId>${BU.PARAMETERS.SalesOrg}</salesOrganisationId>
  <distributionChannelCode>R1</distributionChannelCode>
  <priceZoneId>1</priceZoneId>
  <storeInternalId>9090</storeInternalId>
  <salesPriceLevelCode>1</salesPriceLevelCode>
  <defaultUnitOfMeasure>PCE</defaultUnitOfMeasure>
  <validityFrom>CREATION_DAY</validityFrom>
  <validityTo>INFINITY</validityTo>
</import-config>
```

So this finally results in:

```
<salesOrganisationId>RS10</salesOrganisationId>
```

2.2.2.2 Inheritance and Overriding

If you created a business unit hierarchy, all settings from higher nodes (business units) are inherited by child nodes.

Here we are in a sub business unit of the central business unit named "sap" and you can see that this node inherited the value:

Edit Business Unit ✕

GENERAL

CONFIGURATION

STATISTICS

OPTIMIZATION

PARAMETERS

Parameter
SalesOrg

from sap

Value
RS10

–

ADD SETTING

CANCEL

APPLY

You can override that value on this level for the current business unit and all the children:

Edit Business Unit



GENERAL

CONFIGURATION

STATISTICS

OPTIMIZATION

PARAMETERS

Parameter
SalesOrg

from sap

Value
RS11

overwritten



ADD SETTING

CANCEL

APPLY

2.3 Import/Export Configuration: SAP Passport

2.3.1 Summary

You can add an SAP passport to any HTTP request that you send to SAP backbone services. The passport helps the receiver to separate external systems that count to licensed access or not.

The SAP passport uses special HTTP headers.

2.3.2 Configuration

You can enable the SAP passport for any HTTP request via the import/export XML configuration.

If you enable the SAP passport it is transmitted in the **HTTP header field SAP-PASSPORT**.

2.3.2.1 Parameters

```
<httpResource>
...
  <sapPassport>
    <componentName>XX-PART-GKS</componentName>
    <applicationComponentName>DynamicPricing</applicationComponentName>
    <tenantId>200</tenantId>
    <service>0</service>
    <action>SAPEXportXmlPrices</action>
    <actionType>11</actionType>
  </sapPassport>
</httpResource>
```

Tag	Optional	Default	Value Range	Description
componentName			String(32)	SAP component ID of the partner application
applicationComponentName				Application name
tenantId			String(3)	Tenant to access
service			String(2)	Service number
action			String(40)	Action name
actionType			String(2)	Action number

2.3.2.2 Example

```
<httpResource>
  <resourceId>MultiChannelSalesQuery</resourceId>
  <resourceUrl>http://fiori-dev-hanadb01.gk.gk-
software.com:8000/MultiChannelSalesQuery</resourceUrl>
  <query><?format=xml</query>
  <batchSize>500</batchSize>
  <type>TransactionLog</type>
  <authentication>
    <user>xxxx</user>
    <encryptedPassword>xxxx</encryptedPassword>
  </authentication>
  <sapPassport>
    <componentName>XX-PART-GKS</componentName>
    <applicationComponentName>DynamicPricing</applicationComponentName>
    <tenantId>200</tenantId>
    <service>0</service>
    <action>SAPExportXmlPrices</action>
    <actionType>11</actionType>
  </sapPassport>
</httpResource>
```

Default Values set by the Implementation

Name	Fixed Values	Max Size	Dynamic Values	Example
Eye Catcher	TH			
Version	"3"			
Length			Length of the current header	230
Trace Flags	0x0000			
User ID	empty			
Transaction ID (GUID)	randomUUID	32	Random UUID	
Component Type	"24"			
Root Context ID	randomUUID	16	Random UUID	
Connection ID	randomUUID	16	Random UUID	
Connection Counter	1			
Var Part Counter	0			
Var Part Offset	226			
Eye Catcher	TH			

2.4 Export

2.4.1 Export Protocol

2.4.1.1 Summary

This document describes the configuration options for the import and export protocol. The protocol stores events, warnings and errors. They can be displayed via the UI or they can be downloaded via the UI. They are persisted in the import/export task.

2.4.1.2 Protocols

Output	Type	Description
Application Log	Short	Protocol is logged to application log
Protocol: Displayed for task	Long	Protocol dialog for Errors, warning, events
Protocol: Download via GUI	Long	You can download the protocol as JSON

As you can see, there are two types:

- Short

- Long

For each type there are defaults for the count of events, warnings and errors. Via configuration you can change the defaults.

2.4.1.3 Configuration

You can fine tune, how many events, errors and warnings are reported. The configuration is optional. If you don't add the protocol tag to your configuration, the default values are used.

For debugging you can switch to unlimited messages by just setting the values to -1. All negative values mean unlimited.

For debugging purposes it might be useful to temporarily increase the counts for the messages. Please don't forget to reset the debug configuration later on, as saving tons of messages is at a cost in the cloud. Messages are saved to database and logs which might increase the cost of your instance dramatically.

```
<import-config>
...
<protocol>
  <longProtocolErrorLimit>1000</longProtocolErrorLimit>
  <longProtocolWarningLimit>1000</longProtocolWarningLimit>
  <longProtocolEventLimit>-1</longProtocolEventLimit>
  <shortProtocolErrorLimit>1000</shortProtocolErrorLimit>
  <shortProtocolWarningLimit>1000</shortProtocolWarningLimit>
  <shortProtocolEventLimit>200</shortProtocolEventLimit>
</protocol>
```

Tag	Default	Value Range	Description
longProtocolErrorLimit	512	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of errors reported in the long protocol
longProtocolWarningLimit	512	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of warnings reported in the long protocol
longProtocolEventLimit	1024	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of events reported in the long protocol
shortProtocolErrorLimit	64	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of errors reported in the sort protocol
shortProtocolWarningLimit	64	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of warnings reported in the short protocol
shortProtocolEventLimit	256	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of events reported in the short protocol

2.4.2 Export Communication Interface

2.4.2.1 Summary

This document describes the following communication paths for the export interfaces:

1. File system
2. HTTP endpoint

2.4.2.2 Configuration

Export to the File System

```
<fileOutboundAdapter>
  <outputDir>/tmp/pricing/export</outputDir>
  <isLockFileEnabled>true</isLockFileEnabled>
  <isArchiveToTaskEnabled>false</isArchiveToTaskEnabled>
  <lockFileName>lock</lockFileName>
  <waitOnLockFileFoundInSeconds>5</waitOnLockFileFoundInSeconds>
  <lockFileFoundRetryCount>2</lockFileFoundRetryCount>
  ...
</fileOutboundAdapter>
```

Tag	Child Tag	Value Range	Default	Description
outputDir		String		Directory for output files.
isLockFileEnabled		true, false		true = Use lock file. false = no lock file.
lockFileName		String		name of the lock file. Only used if isLockFileEnabled = true
waitOnLockFileFoundInSeconds		Integer 0..		If job wants to write the lock file and the lock file is already there, then wait some seconds for another try.
lockFileFoundRetryCount		Integer 0..		If job wants to write the lock file and the lock file is already there, then wait some seconds for another try: This parameter controls how many retries until the job finally fails.
isArchiveToTaskEnabled		true, false	false	true = Save the export data also to the database. You can download the data via the GUI after exporting. false = Do not save the export data to the database

Example

Scenario:

- Enable Lock File "lock"
- Try to create lock file. If available, wait for 5 seconds. Try that 3 times, then fail

```
<fileOutboundAdapter>
  <outputDir>/home/export</outputDir>
  <isLockFileEnabled>true</isLockFileEnabled>
  <isArchiveToTaskEnabled>false</isArchiveToTaskEnabled>
  <lockFileName>lock</lockFileName>
  <waitOnLockFileFoundInSeconds>5</waitOnLockFileFoundInSeconds>
  <lockFileFoundRetryCount>3</lockFileFoundRetryCount>
  <outboundDataMappings>
    <outboundDataMapping>
      <mappingId>OptimizedPrice</mappingId>
      <type>OptimizedPrice</type>
      <outputName>prices.csv</outputName>
    </outboundDataMapping>
  </outboundDataMappings>
</fileOutboundAdapter>
```

Example:

1. Export is started by scheduler at a given time
2. Exporter tries to write "home/export/lock". If file is already available, then wait 5 seconds and try to write lock file again. If this fails, try it 3 times altogether. After that: fail
3. Export data
4. Remove lock file

Definition of Output Files

Several data mappings can be defined within the group outboundDataMappings, each with its own outboundDataMapping.

Each of these mappings can later be identified by the Export Adapter via the type and the mappingId, and assigned to the respective output data stream.

Example:

type = OptimizedPrice

The Export Adapter searches the outboundDataMapping of the type "OptimizedPrice" for the prices and then gets all the settings, e.g. the file name (outputName).

Several mappings can be defined per type, in this case the mappingIds must be different. Generally, the export must be ready for ID and type.

outboundDataMapping

Tag	Child Tag	Value Range	Description
mappingId		String	Unique ID for a mapping rule: When the mapping rule is applied (the first suitable rule is used), the ID of the adapter can be used to determine which rule to use.
type		String	Adapters filter by this data type. Therefore, there is a connection between adapters and the mappings. Example: A Price Export Adapter searches the "OptimizedPrice". It gets the file name from the outputName tag.
outputName		String	File names of the file to be exported of a type. The outputName can contain dynamic content. Supported parameters are: <ul style="list-style-type: none"> • \${DATE} <ul style="list-style-type: none"> ○ Export the export date in the format YYYYMMDD, e.g. 20191001 • \${TIME} <ul style="list-style-type: none"> ○ Export the export time in the format HHMMSS, e.g. 12:32:11 • \${BU} <ul style="list-style-type: none"> ○ business unit id

Export to an HTTP Endpoint

```

<httpOutboundAdapter>
  <waitOnFailedInSeconds>5</waitOnFailedInSeconds>
  <isArchiveToTaskEnabled>false</isArchiveToTaskEnabled>
  <retryCounter>0</retryCounter>
  <httpResources>
    <httpResource>
      <resourceId>OptimizedPrice</resourceId>
      <resourceUrl>http://test.de</resourceUrl>
      <query></query>
      <type>OptimizedPrice</type>
      <authentication>
        <user>test</user>
        <encryptedPassword>12344</encryptedPassword>
      </authentication>
    </httpResource>
  </httpResources>
</httpOutboundAdapter>

```

Code Block 1 Sample configuration

Tag	Mandatory Field	Value Range	Default	Description
waitOnFileValidationFailedInSeconds	x	[0..Max Integer]		Number of seconds to wait when the connection attempt has failed. After that, the validation is repeated.
isArchiveToTaskEnabled		true, false	false	true = Save the export data also to the database. You can download the data via the GUI after exporting. false = Do not save the export data to the database
retryCounter	x	[0..Max Integer]		Number of repeat attempts when a connection attempt has failed. 0 means cancel after first error.

HTTP Resource

Tag	Child Tag	Mandatory Field	Value Range	Default	Description
httpMethod			POST, PUT, GET	POST	HTTP method

For the rest please see details in the Import Communication.

Export to a S/FTP Endpoint

This adapter supports both FTP and SFTP connections. You need to configure the type and ports accordingly.

```
<sftpOutboundAdapter>
  <outboundType>SFTP</outboundType>
  <host>192.168.0.2</host>
  <port>22</port>
  <user>user</user>
  <encryptedPassword>password</encryptedPassword>
  <sftpConnectionTimeout>30</sftpConnectionTimeout>
  <remoteExportDir>.</remoteExportDir>
  <waitOnFailedInSeconds>5</waitOnFailedInSeconds>
  <outboundDataMappings>
    <outboundDataMapping>
      <mappingId>OptimizedPrice</mappingId>
      <type>OptimizedPrice</type>
      <outputName>prices.csv</outputName>
    </outboundDataMapping>
  </outboundDataMappings>
</sftpOutboundAdapter>
```

Code Block 2 Sample configuration

Tag	Mandatory Field	Value Range	Default	Description
outboundType	x	FTP, SFTP		FTP for accessing FTP server, SFTP for SFTP server
host	x	String		IP address of the SFTP server.
port	x	Integer		Port of the S/FTP server. Usually 22 for SFTP, port 21 for FTP
sftpConnectionTimeout		[0..Max Integer]	45	Timeout in Seconds of the SFTP connection.
remoteExportDir	x	String		Directory of the SFTP server to which the files will be exported.
waitOnFailedInSeconds	x	[0..Max Integer]		Number of seconds to wait when the connection attempt has failed. The call is restarted after that.
retryCounter		[0..Max Integer]	5	Number of repeat attempts when a connection attempt has failed. 0 means cancel after first error.
isArchiveToTaskEnabled		true,false	false	Backup of the input data in the archive. Data can be called later via the GUI.

Definition of Output Files

Several data mappings can be defined within the group `outboundDataMappings`, each with its own `outboundDataMapping`.

Each of these mappings can later be identified by the Export Adapter via the type and the `mappingId`, and assigned to the respective output data stream.

The Export Adapter searches the `outboundDataMapping` of the type "OptimizedPrice" for the prices and then gets all the settings, e.g. the file name (`outputName`).

Several mappings can be defined per type, in this case the `mappingIds` must be different. Generally, the export must be ready for ID and type.

outboundDataMapping

Tag	Child Tag	Value Range	Description
mappingId		String	Unique ID for a mapping rule: When the mapping rule is applied (the first suitable rule is used), the ID of the adapter can be used to determine which rule to use.
type		String	Adapters filter by this data type. Therefore, there is a connection between adapters and the mappings. Example: A Price Export Adapter searches the "OptimizedPrice". It gets the file name from the outputName tag.
outputName		String	File names of the file to be exported of a type. The outputName can contain dynamic content. Supported parameters are: <ul style="list-style-type: none"> • \${DATE} <ul style="list-style-type: none"> ○ Export the export date in the format YYYYMMDD, e.g. 20191001 • \${TIME} <ul style="list-style-type: none"> ○ Export the export time in the format HHMMSS, e.g. 12:32:11 • \${BU} <ul style="list-style-type: none"> ○ business unit id

Authentication

For more information see in the Import Configuration.

2.4.2.3 Download Export Data via REST Interface (PULL)

Please see details here.

2.4.3 Export CSV Writing

2.4.3.1 Summary

This document describes the configuration details about writing CSV data.

2.4.3.2 Configuration of exporting CSV Data

XML Structure

```
<csvWriter>
  <charset>UTF-8</charset>
  <writeHeader>true</writeHeader>
  <escapeMode>BACKSLASH</escapeMode>
  <endOfLineMode>UNIX</endOfLineMode>
  <columnSeparator>|</columnSeparator>
  <textQualifier>"</textQualifier>
</csvWriter>
```

csvWriter

Tag	Optional	Default	Value Range	Description
charset	x	UTF-8	Charset Name	Charset used for writing data to the CSV
writeHeader	x	true	true, false	Should the header be written
escapeMode	x	BACKSLASH	BACKSLASH EXCEL	BACKSLASH: Escape with \ EXCEL: Escape with ""
endOfLineMode	x	UNIX	UNIX, WINDOWS	UNIX: /n WINDOWS: /r/n
columnSeparator	x		Character	Character to be used to separate cells
textQualifier	x	not used	Character	? If you use this tag, a text qualifier is set. Omit it if you don't need it.

Example

```
<csvWriter>
  <charset>UTF-8</charset>
  <writeHeader>true</writeHeader>
  <escapeMode>BACKSLASH</escapeMode>
  <endOfLineMode>WINDOWS</endOfLineMode>
  <columnSeparator>|</columnSeparator>
</csvWriter>
```

2.4.4 Export Timezones

2.4.4.1 Summary

Some exporters can handle timezones. The dynamic pricing application stores date/time in UTC. Exporters that can handle the export to a configured timezone explicitly document that in their interface description.

If configuration is supported, there are several ways to configure the time zone to be used for the export:

1. Pass the timezone in the export request
2. Configure a timezone in the configuration file
3. Configure a timezone in the business unit

The priority is from 1 to 3. So if you pass a timezone to the request, this one is used instead of anyone you've configured in the configuration file or business unit.

2.4.4.2 Timezone in the Request

You can pass a timezone using the parameter "**timezone**" in your export triggering request:

```
http://...../?timezone=UTC
```

See here for more details about the format: <https://docs.oracle.com/javase/8/docs/api/java/time/ZoneId.html#of-java.lang.String->

Please make sure your timezone is URL encoded, e.g. GMT+02:00 results in **GMT%2B02:00**

2.4.4.3 Timezone in the XML configuration

```
<import-config>
  <timeZone>GMT+02:00</timeZone>
  ...
</import-config>
```

The format resembles the one stated in the chapter about the request, except the URL encoding.

2.4.4.4 Timezone in the Business Unit

You can maintain a business unit specific timezone in the business unit's edit screen:

Edit Business Unit



GENERAL

SETTINGS

OPTIMIZATION

Globalization

Master currency
EUR

☒ Activate rounding

☒ Cash rounding

Scheduler time zone
GMT+01:00

2.4.5 Export Pipelines

2.4.5.1 Summary

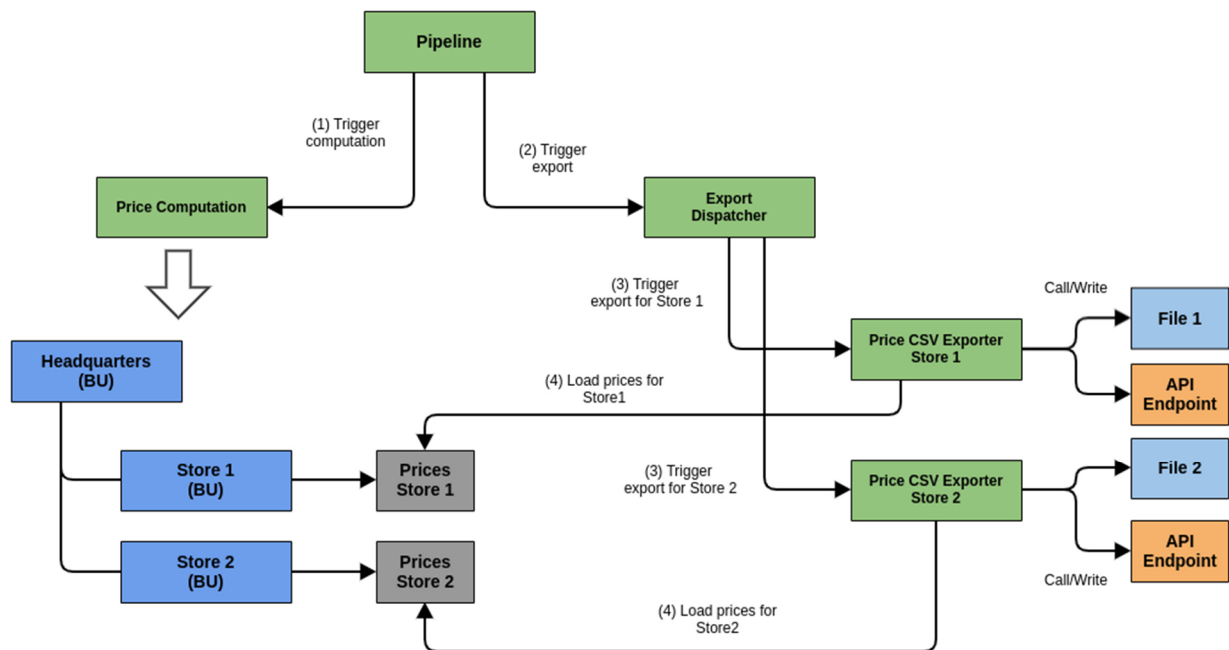
With a pipeline for exporters you can:

1. Run multiple exporters in a sequence
2. Export optimized prices of a business unit hierarchy for each business unit to one/variable endpoint or multiple files
3. Export optimized prices of a business unit hierarchy to one endpoint or file

To run more different exporters in sequence, you just have to add them to a new pipeline using the GUI. For the other use cases you need additional configuration in your exporter XML configuration.

2.4.5.2 Export optimized prices of a business unit hierarchy for each business unit to one/variable endpoint or multiple files

Overview



Configuration

- Start with creating a business unit hierarchy:
 - Level 1: Headquarters (id = hq), disable price computation
 - Level 2: Store 1 (id = store1)
 - Level 2: Store 2 (id = store2)
- Then you need to create a segment and a computation. For both elements, enable the setting to be a master for the hierarchy.
- Link the segment to the computation.
- Create a pipeline and link the computation
- Create an exporter (and enable master flag for hierarchy) and link it to the pipeline too.
- Configure the exporter with XML configuration. See details about the sections to modify below.

Orchestration

```

<orchestration>
  <exportMode>ONE_STREAM_FOR_BU</exportMode>
</orchestration>

```

Tag	Value Range	Description
exportMode	ONE_STREAM_FOR_BU	Each business unit just reads

File Export

You can use file or HTTP export. Example for file export:

```

<fileOutboundAdapter>
  <outputDir>/tmp/export/${BU}</outputDir>
  <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  <isLockFileEnabled>>false</isLockFileEnabled>
  <lockFileName>lock</lockFileName>
  <waitOnLockFileFoundInSeconds>5</waitOnLockFileFoundInSeconds>
  <lockFileFoundRetryCount>2</lockFileFoundRetryCount>
  <outboundDataMappings>
    <outboundDataMapping>
      <mappingId>OptimizedPrice</mappingId>
      <type>OptimizedPrice</type>
      <outputName>${DATE}_${TIME}_prices.csv</outputName>
    </outboundDataMapping>
  </outboundDataMappings>
</fileOutboundAdapter>

```

Two tags support special placeholders:

Tag	Description	Example
outputDir	Folder for the output file. Placeholders: <ul style="list-style-type: none"> • \${BU} = business unit id 	/tmp/export/\${BU}
outputName	Filename to output the prices. Placeholders: <ul style="list-style-type: none"> • \${BU} = business unit id • \${DATE} = current date (format: YYYYMMDD) • \${TIME} = current time (format: HHMMSS) 	\${DATE}_\${TIME}_prices.csv

HTTP Export

```

<httpOutboundAdapter>
  <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  <waitOnFailedInSeconds>5</waitOnFailedInSeconds>
  <retryCounter>2</retryCounter>
  <httpResources>
    <httpResource>
      <resourceId>OptimizedPrice</resourceId>
      <resourceUrl>http://test.de?organisation=${BU}</resourceUrl>
      <httpMethod>POST</httpMethod>
      <type>OptimizedPrice</type>
      <authentication>
        <user>test</user>
        <encryptedPassword>test</encryptedPassword>
      </authentication>
    </httpResource>
  </httpResources>
</httpOutboundAdapter>

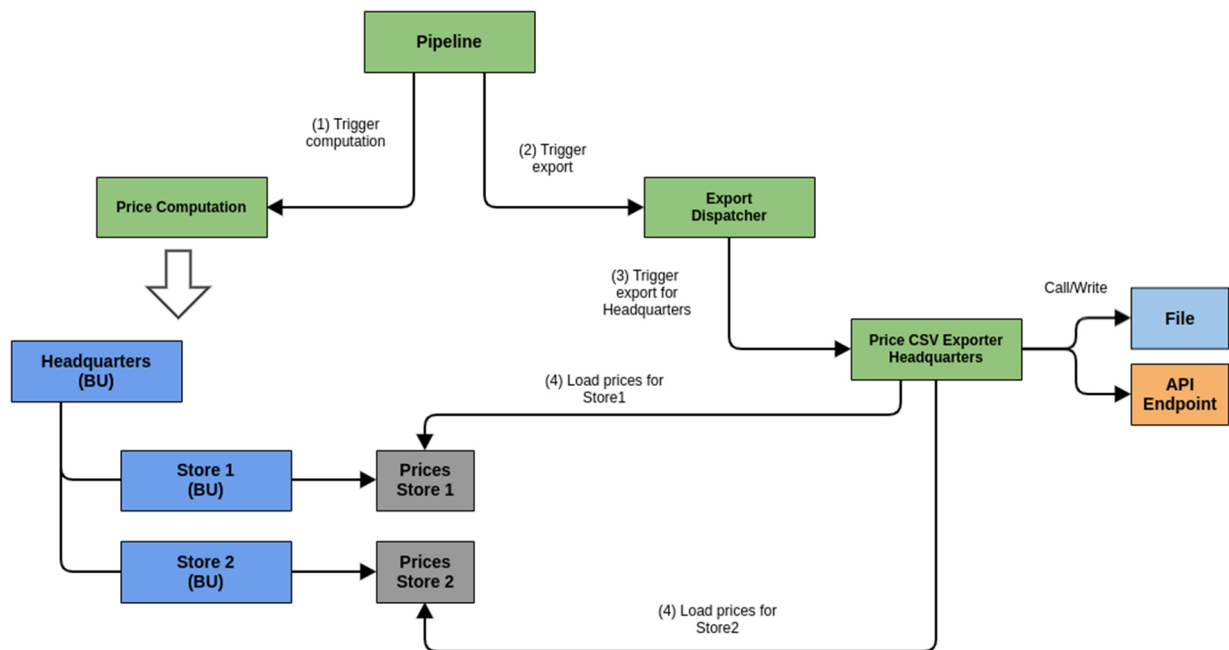
```

There is one tag that supports extended configuration for the pipeline:

Tag	Description	Example
resourceUrl	URL to post data to. Placeholders: <ul style="list-style-type: none"> • \${BU} = business unit id 	http://test.de?organisation=\${BU}

2.4.5.3 Export optimized prices of a business unit hierarchy to one endpoint or file

Overview



Configuration

1. Start with creating a business unit hierarchy:
 - a. Level 1: Headquarters (id = hq), disable price computation
 - b. Level 2: Store 1 (id = store1)
 - c. Level 2: Store 2 (id = store2)
2. Then you need to create a segment and a computation. For both elements, enable the setting to be a master for the hierarchy.
3. Link the segment to the computation.
4. Create a pipeline and link the computation
5. Create an exporter (and enable master flag for hierarchy) and link it to the pipeline too.
6. Configure the exporter with XML configuration. See details about the sections to modify below.

Orchestration

```
<orchestration>
  <exportMode>ONE_STREAM_FOR_ALL_BUS</exportMode>
</orchestration>
```

Tag	Value Range	Description
exportMode	ONE_STREAM_FOR_ALL_BUS	Read prices from each BU and put it to a common output

File Export

You can use file or HTTP export. Example for file export:

```

<fileOutboundAdapter>
  <outputDir>/tmp/export/${BU}</outputDir>
  <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  <isLockFileEnabled>false</isLockFileEnabled>
  <lockFileName>lock</lockFileName>
  <waitOnLockFileFoundInSeconds>5</waitOnLockFileFoundInSeconds>
  <lockFileFoundRetryCount>2</lockFileFoundRetryCount>
  <outboundDataMappings>
    <outboundDataMapping>
      <mappingId>OptimizedPrice</mappingId>
      <type>OptimizedPrice</type>
      <outputName>${DATE}_${TIME}_prices.csv</outputName>
    </outboundDataMapping>
  </outboundDataMappings>
</fileOutboundAdapter>

```

Two tags support special placeholders:

Tag	Description	Example
outputDir	Folder for the output file. Placeholders: <ul style="list-style-type: none"> • \${BU} = business unit id 	/tmp/export/\${BU}
outputName	Filename to output the prices. Placeholders: <ul style="list-style-type: none"> • \${BU} = business unit id • \${DATE} = current date (format: YYYYMMDD) • \${TIME} = current time (format: HHMMSS) 	\${DATE}_\${TIME}_prices.csv

HTTP Export

```

<httpOutboundAdapter>
  <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  <waitOnFailedInSeconds>5</waitOnFailedInSeconds>
  <retryCounter>2</retryCounter>
  <httpResources>
    <httpResource>
      <resourceId>OptimizedPrice</resourceId>
      <resourceUrl>http://test.de?organisation=${BU}</resourceUrl>
      <httpMethod>POST</httpMethod>
      <type>OptimizedPrice</type>
      <authentication>
        <user>test</user>
        <encryptedPassword>test</encryptedPassword>
      </authentication>
    </httpResource>
  </httpResources>
</httpOutboundAdapter>

```

There is one tag that supports extended configuration for the pipeline:

Tag	Description	Example
resourceUrl	URL to post data to. Placeholders: <ul style="list-style-type: none"> • \${BU} = business unit id 	http://test.de?organisation=\${BU}

2.4.5.4 Download Export Data via REST Interface (PULL)

The exporters support both HTTP and File push interfaces to upload optimized prices. Besides that there is also a pull interface:

```
<PIPELINE REST API>?action=getexportdata
```

Just use the pipeline REST API (GUI: open the pipeline and open the details) and add ?action=getexportdata.
Example:

```
https://test.com/air/admin/res/gk/plugins/exec/prudsys/prudsys/pricing/orchestratedpricingservice/integration/computation?action=getexportdata
```

This returns a zip file with all exported data of the last successful export. The folder structure looks like this:

- <internal name of the export service>
 - <id of the export service>
 - <id of the business unit>
 - data

Example:

- price_sapes_exportservice
 - sappriceexport
 - integration
 - 7743894994999994

It's recommended to provide custom IDs for the business units and the export services. Don't use the defaults, which are random UUIDs.

2.4.5.5 Additional Configuration Options using Business Unit Parameters

Besides the configuration options mentioned above, the configuration file also supports business unit parameters that are injected at any tag of your configuration file.

For details, please see the section "Import/Export Configuration: Business Unit Parameters" in the Configuration Guide.

2.4.6 Prices Export

2.4.6.1 Prices Export - CSV

Summary

This export is used to export optimized prices in the **CSV data format**. The export may generate a file or transmit the data to an HTTP endpoint via POST.

The export also supports the conversion of the local currency of the business unit to another currency. The exchange rate must have been imported in advance, otherwise no price will be exported.

Communication

The following options are supported for the communication:

Trigger	Description
GET	• The result of the last successful calculation run is exported (as file or to an endpoint)
End of the price calculation	• The result of the current calculation run is exported (as file or to an endpoint)

Output Data

Example

```
businessUnit|exportTimeStamp|itemId|optimizedPrice|currency|origin|manualPrice|author|overrideTimeStamp
ecommerce|2019-02-01 12:41:11 GMT+2|1277474|1.14|EUR|S|||
ecommerce|2019-02-01 12:41:11 GMT+2|77475|1.20|EUR|M|1.15|mope|2018-11-01 12:22:11 GMT+2
```

Mapping and Validation

CSV field	Format	Example	Mandatory field	Description
businessUnit	String	shop999	x	To which business unit should the prices be applicable?
exportTimeStamp	yyyy-MM-DD HH:mm:ss	2019-02-01 12:44:11 GMT+2	x	Timestamp of the export. Start of export, same for all exported items of one export.
itemId	String	334556	x	Item for which a price has been determined
optimizedPrice	Monetary amount: US format with a maximum of two decimal places, no thousands separator, "." for comma	1.15	x	Optimized price
currency	ISO 4217*	USD	x	Currency of the price
origin	M, S	S	x	M = Manually applied price (masterdata override) S = Calculated by a pricing strategy
manualPrice	Monetary amount: US format with a maximum of two decimal places, no thousands separator, "." for comma	1.14		If origin = M or R, then manually changed price
author	String	maxx		If origin = M, then user name of the person who changed the price
overrideTimeStamp	yyyy-MM-dd HH:mm:ss O	2018-12-14 12:33:21 GMT+2		If origin = M, then the timestamp of the manual price change.

*) ISO 4217 for currency code: https://de.wikipedia.org/wiki/ISO_4217

Handling Timezones

The "exportTimeStamp" and "overrideTimeStamp" columns are exported for a given timezone. You can tell the exporter which timezone to use to convert it from the server's UTC to another timezone in this priority:

1. Pass the timezone parameter to the GET request
2. Configure a timezone in the XML configuration
3. Define a timezone in the business unit (default is UTC)

See more details in the chapter "Export Timezones".

Example:

Let's imagine you configured the timezone GMT+02:00 in the business unit. The exporter then exports:

```
ecommerce|2019-02-01 12:41:11 GMT+2|77475|1.20|EUR|M|1.15|mope|2018-11-01 12:22:11 GMT+2
```

The timezone setting is also used for the TIME and DATE parameters that can be used for the export file name.

Configuration

General

```
<priceExportMode>ALL</priceExportMode>
<businessUnit>default</businessUnit>
<exchangeToCurrency>EUR</exchangeToCurrency>
```

Tag	Child Tag	Value Range	Mandatory field	Default	Description
priceExportMode		ALL ONLY_CHANGED		ALL	<ul style="list-style-type: none"> ALL = Export all prices ONLY_CHANGED = Export only changed prices
businessUnit		String			If filled, this tag will fill the first column businessUnit in the export. If left out, the export uses the identifier of the current business unit (i.e. the one where the export is executed).
exchangeToCurrency		Currency Code ISO 4217			Currency code of the currency to convert the price. There must be defined a conversion rate for business unit master currency to this currency (using exchange rate importer to import that rate)
timeZone		See chapter "Export Timezones". E.g. GMT+02:00			Timezone to use for the time that is exported in exportTimeStamp and overrideTimeStamp and the filename parameters DATE and TIME

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Export to File System

See "Export Communication Interface".

Export to HTTP Endpoint

For the general description, see "Export Communication Interface".

CSV Writing

Please see details about the configuration here.

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <priceExportMode>ALL</priceExportMode>
  <businessUnit>9090</businessUnit>
  <csvWriter>
    <charset>UTF-8</charset>
    <writeHeader>true</writeHeader>
    <escapeMode>BACKSLASH</escapeMode>
    <columnSeparator>|</columnSeparator>
  </csvWriter>
  <fileOutboundAdapter>
    <outputDir>/tmp/pricing/export</outputDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <waitOnLockFileFoundInSeconds>5</waitOnLockFileFoundInSeconds>
    <lockFileFoundRetryCount>2</lockFileFoundRetryCount>
    <outboundDataMappings>
      <outboundDataMapping>
        <mappingId>OptimizedPrice</mappingId>
        <type>OptimizedPrice</type>
        <outputName>prices.csv</outputName>
      </outboundDataMapping>
    </outboundDataMappings>
  </fileOutboundAdapter>
</import-config>
```

Code Block 3 Sample configuration

2.4.6.2 Prices Export - SAP Enterprise Service

Summary

This export is used to export optimized prices in **XML format to an SAP ERP system, via the Enterprise Service `SalesPriceSpecCalcERPCreateRequestConfirmation_In`**.

The export may generate a file or transmit the data to the SAP HTTP endpoint via POST.

The export also supports the conversion of the local currency of the business unit to another currency. The exchange rate must have been imported in advance, otherwise no price will be exported.

You can export prices to a specific organisational level, e.g. distribution channel and sales organisation but also for a given store.

Communication

The following options are supported for the communication:

Trigger	Description
GET	<ul style="list-style-type: none">The result from the last successful calculation run is exported (as file or to an SAP endpoint)
End of the price calculation	<ul style="list-style-type: none">The result from the current calculation run is exported (as file or to an SAP endpoint)

SAP Customizing

Please see the Integration Guide for more information about necessary customizing.

Output Data

Example

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <SOAP-ENV:Body>
    <n0:SalesPriceSpecCalcERPCreateRequest_sync xmlns:n0="http://sap.com/xi/SAPGlobal20/Global">
      <MessageHeader>
        <UUID>307e8f93-9939-49e4-a685-5b7bbee4464e </UUID>
      </MessageHeader>
      <SalesPriceSpecificationCalculation>
        <Item>
          <ProductInternalID>B31231010000</ProductInternalID>
          <MeasureUnitCode>PCE</MeasureUnitCode>
          <SalesOrganisationID>RS10</SalesOrganisationID>
          <DistributionChannelCode>R1</DistributionChannelCode>
          <ValidityPeriod>
            <IntervalBoundaryTypeCode>3</IntervalBoundaryTypeCode>
            <StartTimePoint>
              <TypeCode>1</TypeCode>
              <Date>2019-04-05</Date>
            </StartTimePoint>
            <EndTimePoint>
              <TypeCode>1</TypeCode>
              <Date>9999-12-31</Date>
            </EndTimePoint>
          </ValidityPeriod>
          <Price>
            <SpecificationElementPropertyDefinitionClassCode>2</SpecificationElementPropertyDefinitionClassCode>
            <Price>
              <Amount currencyCode="USD">18.55</Amount>
              <BaseQuantity>1</BaseQuantity>
              <BaseQuantityTypeCode>PCE</BaseQuantityTypeCode>
            </Price>
          </Price>
        </Item>
      </SalesPriceSpecificationCalculation>
    </n0:SalesPriceSpecCalcERPCreateRequest_sync>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Mapping and Validation

XML field	Format	Example	Mandatory field	Description
ProductInternalID	String	334556	x	Item for which a price has been calculated
MeasureUnitCode	String	PCE	x	Unit of measure of the item. Item.MeasureUnitCode is used. If not available, the default code from configuration is used.
SalesOrganisationID	String	RS10	x	Addresses the price level 1, 2, 3
DistributionChannelCode	String	R1	x	Addresses the price level 1, 2, 3
PriceZoneID	String			Addresses the price level 2
StoreInternalID	String			Addresses the price level 3
IntervalBoundaryTypeCode	Integer	3	x	Fixed 3 = interval including start and end timepoint [start date, end date]
StartTimePoint.TypeCode	Integer	1	x	Fixed 1 = Date
StartTimePoint.Date	Date YYYY-MM-DD	2019-01-03	x	Validity start date of the price
EndTimePoint.TypeCode	Integer	1	x	Fixed: 1 = Date
EndTimePoint.Date	Date YYYY-MM-DD	2019-01-03	x	Validity end date of the price
Price.SpecificationElementPropertyDefinitionClassCode	Integer	2	x	Fixed: 2 = Price Optimization
Price.Amount	Monetary amount	1.15	x	Optimized price
Price.Amount.CurrencyCode	ISO 4217*	USD	x	Currency of the price
Price.BaseQuantity	Integer	1	x	Fixed: 1 = 1 quantity of UOM
Price.BaseQuantityTypeCode	String	PCE	x	Unit of measure of the item. Item.MeasureUnitCode is used. If not available, the default code from configuration is used.

*) ISO 4217 for currency code: https://de.wikipedia.org/wiki/ISO_4217

Handling Timezones

You can tell the exporter which timezone to use to convert the export time in UTC to another timezone in this priority:

1. Pass the timezone parameter to the GET request
2. Configure a timezone in the XML configuration
3. Define a timezone in the business unit (default is UTC)

See more details in the chapter "Export Timezones".

The timezone setting is used for the TIME and DATE parameters that can be used for the export file name.

Configuration

General

```
<priceExportMode>ALL</priceExportMode>
<salesOrganisationId>RS10</salesOrganisationId>
<distributionChannelCode>R1</distributionChannelCode>
<priceZoneId>1</priceZoneId>
<storeInternalId>9090</storeInternalId>
<salesPriceLevelCode>3</salesPriceLevelCode>
<defaultUnitOfMeasure>PCE</defaultUnitOfMeasure>
<validityFrom>CREATION_DAY</validityFrom>
<validityTo>INFINITY</validityTo>
<exchangeToCurrency>EUR</exchangeToCurrency>
```

Tag	Child Tag	Value Range	Mandatory field	Default	Description
priceExportMode		ALL ONLY_CHANGED		ALL	<ul style="list-style-type: none">ALL = Export all pricesONLY_CHANGED = Export only changed prices
salesOrganisationId		String	x		Addresses the price level 1, 2, 3
distributionChannelCode		String	x		Addresses the price level 1, 2, 3
priceZoneId		String			Addresses the price level 2
storeInternalId		String			Addresses the price level 3
salesPriceLevelCode		Integer Product: 1-3	x		Price Level. The product supports level 1-3. All other levels are treated as custom level. For each level the other parameters have to be set (see above): 1 = SalesOrganisation + DistributionChannel 2 = SalesOrganisation + DistributionChannel + PriceZone 3 = SalesOrganisation + DistributionChannel + Store
defaultUnitOfMeasure		String	x		DefaultUOM, used if not set in item master data
validityFrom		CREATION_DAY, NEXT_DAY	x		Defining the exported start validity date of the optimized price: CREATION_DAY = Current day of export NEXT_DAY = Next day relative to export date
validityTo		INFINITY	x		Results in date 12-31-9999
exchangeToCurrency		Currency Code ISO 4217			Currency code of the currency to convert the price. There must be defined a conversion rate for business unit master currency to this currency (using exchange rate importer to import that rate)
timeZone		See chapter "Export Timezones". E.g. GMT+02:00			Timezone to use for the file name parameters DATE, TIME

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during export.

Export to File System

See "Export Communication Interface".

Export to HTTP endpoint

For the general description, see "Export Communication Interface".

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <salesOrganisationId>R10</salesOrganisationId>
  <distributionChannelCode>R1</distributionChannelCode>
  <priceZoneId>1</priceZoneId>
  <storeInternalId>9090</storeInternalId>
  <!-- 1 = DISTRIBUTION_CHAIN, 2 = PRIZE_ZONE, 3 = STORE, any other = custom level -->
  <salesPriceLevelCode>1</salesPriceLevelCode>
  <defaultUnitOfMeasure>PCE</defaultUnitOfMeasure>
  <!-- CREATION_DAY, NEXT_DAY -->
  <validityFrom>CREATION_DAY</validityFrom>
  <!-- INFINITY -->
  <validityTo>INFINITY</validityTo>
  <httpOutboundAdapter>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <waitOnFailedInSeconds>5</waitOnFailedInSeconds>
    <retryCounter>0</retryCounter>
    <httpResources>
      <httpResource>
        <resourceId>OptimizedPrice</resourceId>
        <resourceUrl>http://test.de</resourceUrl>
        <query></query>
        <httpMethod>POST</httpMethod>
        <type>OptimizedPrice</type>
        <authentication>
          <user>test</user>
          <encryptedPassword>test</encryptedPassword>
        </authentication>
      </httpResource>
    </httpResources>
  </httpOutboundAdapter>
</import-config>
```

2.5 Import

2.5.1 Import Protocol

2.5.1.1 Summary

This document describes the configuration options for the import and export protocol. The protocol stores events, warnings and errors. They can be displayed via the UI or they can be downloaded via the UI. They are persisted in the import/export task.

2.5.1.2 Protocols

Output	Type	Description
Application Log	Short	Protocol is logged to application log
Protocol: Displayed for task	Long	Protocol dialog for Errors, warning, events
Protocol: Download via GUI	Long	You can download the protocol as JSON

As you can see, there are two types:

- Short

- Long

For each type there are defaults for the count of events, warnings and errors. Via configuration you can change the defaults.

2.5.1.3 Configuration

You can fine tune, how many events, errors and warnings are reported. The configuration is optional. If you don't add the protocol tag to your configuration, the default values are used.

For debugging you can switch to unlimited messages by just setting the values to -1. All negative values mean unlimited.

For debugging purposes it might be useful to temporarily increase the counts for the messages. Please don't forget to reset the debug configuration later on, as saving tons of messages is at a cost in the cloud. Messages are saved to database and logs which might increase the cost of your instance dramatically.

```
<import-config>
...
<protocol>
  <longProtocolErrorLimit>1000</longProtocolErrorLimit>
  <longProtocolWarningLimit>1000</longProtocolWarningLimit>
  <longProtocolEventLimit>-1</longProtocolEventLimit>
  <shortProtocolErrorLimit>1000</shortProtocolErrorLimit>
  <shortProtocolWarningLimit>1000</shortProtocolWarningLimit>
  <shortProtocolEventLimit>200</shortProtocolEventLimit>
  <isAggregatedStreamStatisticEnabled>true</isAggregatedStreamStatisticEnabled>
</protocol>
```

Tag	Default	Value Range	Description
longProtocolErrorLimit	512	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of errors reported in the long protocol
longProtocolWarningLimit	512	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of warnings reported in the long protocol
longProtocolEventLimit	1024	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of events reported in the long protocol
shortProtocolErrorLimit	64	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of errors reported in the sort protocol
shortProtocolWarningLimit	64	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of warnings reported in the short protocol
shortProtocolEventLimit	256	- MAX_INTEGER, + MAX_INTEGER Any negative value means: + MAX_INTEGER	Maximum of events reported in the short protocol
isAggregatedStreamStatisticEnabled	false	true, false	If true, then all file/stream statistics are aggregated to one statistic. This is done per entity type (e.g. Transactions). This is helpful if you import zip files with thousands of files.

2.5.2 Import Communication Interface

2.5.2.1 Summary

This document describes the following communication paths for the import interfaces:

1. File system
2. HTTP endpoint

2.5.2.2 Configuration

Import from File System

Structure of the File Names

```
<inboundDataMappings>
  <inboundDataMapping>
    <mappingId>Item</mappingId>
    <regExpression>items.*csv</regExpression>
    <type>Item</type>
    <defaultName>items.csv</defaultName>
    <isMandatory>true</isMandatory>
  </inboundDataMapping>
  <inboundDataMapping>
    <mappingId>Category</mappingId>
    <regExpression>categories.*csv</regExpression>
    <type>Category</type>
    <defaultName>category.csv</defaultName>
  </inboundDataMapping>
</inboundDataMappings>
```

Definition of Import Files

Several data mappings can be defined within the inboundDataMappings group, each with its own inboundDataMapping.

Each of these mappings can later be identified by the Import Adapter via type and mappingId, and assigned to the respective input data stream.

Example:

type = Item

The Import Adapter compares the file name of the input stream with the regular expression "items.*csv". If these match, the mapping of the type "Item" is assigned to this file name.

The Import Adapter then gets all the settings for the item data stream, e.g. the ID, default name, etc. Then, the transformers internally search the mappings via the type "Item".

Several mappings can be defined per type, in this case the mappingIds must be different. Generally, the import must be ready for ID and type.

inboundDataMapping

Tag	Child Tag	Default	Value Range	Description
mappingId			String	Unique ID for a mapping rule: When the mapping rule is applied (the first suitable rule is used), the ID in the transformer can be used to determine which rule to use.
regExpression			Regular expression	Regular expression that must be fulfilled for the rule to be applied.
type			String	Transformers filter by these file types. Therefore, there is a connection between transformers and the data found by the rule. Example: An item transformer searches files of the mapping type "Item".
defaultName			String	Default file name. Is used when the file name cannot be determined (e.g. HTTP Post from individual files). This name is transmitted as stream name when the file name cannot be determined, otherwise the file name.
isMandatory		false	true, false	Is the file mandatory? If it is mandatory and not available, then import fails

Configuration

```

<fileInboundAdapter>
  <importDir>/home/import</importDir>
  <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  <copyOption>NONE</copyOption>
  <isLockFileEnabled>false</isLockFileEnabled>
  <lockFileName>lock</lockFileName>
  <triggerOption>NO_TRIGGER</triggerOption>
  <triggerFileName>trigger</triggerFileName>
  <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
  <fileValidationRetryCount>0</fileValidationRetryCount>
</fileInboundAdapter>

```

Tag	Child Tag	Value Range	Description
importDir		String	Directory for input data. The tag copyOption is used to determine what should happen to these files.
isArchiveToTaskEnabled		true, false	Backup of the input data in the archive. Data can be called later via the GUI.
copyOption		NONE, MOVE, COPY	<ul style="list-style-type: none"> NONE: The files remain in the import directory. MOVE: The files are moved from the import directory to the work directory. COPY: The files are copied from the import directory to the work directory.
isLockFileEnabled		true, false	true: <ol style="list-style-type: none"> Start only if there is no lock file (name see parameter lockFileName) Write lock file Copy data to internal folder Remove lock file false: lock file handling is disabled.
lockFileName		String	Name of the lock file to use for locking the import.
triggerOption		NO_TRIGGER WAIT_FOR_TRIGGER CHECK_TRIGGER	<ul style="list-style-type: none"> NO_TRIGGER: Do not check trigger file. A check is carried out to establish if the files exist and if so, import starts. WAIT_FOR_TRIGGER: Wait for a trigger file. If it exists, the files are imported and the trigger file is deleted after that. CHECK_TRIGGER: Check if trigger file exists. If not, the process is canceled.
triggerFileName		String	File name of the trigger file. Only relevant for the triggerOption settings WAIT_FOR_TRIGGER and CHECK_TRIGGER
waitOnFileValidationFailedInSeconds		[0..Max Integer]	Number of seconds to wait when not all the required files are found in the import directory. After that, the validation is repeated. This option has effect only if triggerOption = WAIT_FOR_TRIGGER
fileValidationRetryCount		[0..Max Integer]	Number of repeat attempts when a read attempt has failed. 0 means immediate cancelation after the first failed read operation.

Example

Scenario:

- Enable Trigger File
- Enable Lock File
- Copy data from external folder to internal work folder
- Check once for the trigger file

```
<fileInboundAdapter>
  <importDir>/home/import</importDir>
  <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  <copyOption>COPY</copyOption>
  <isLockFileEnabled>true</isLockFileEnabled>
  <lockFileName>lock</lockFileName>
  <triggerOption>CHECK_TRIGGER</triggerOption>
  <triggerFileName>trigger</triggerFileName>
  <waitOnFileValidationFailedInSeconds>1</waitOnFileValidationFailedInSeconds>
  <fileValidationRetryCount>0</fileValidationRetryCount>
</fileInboundAdapter>
```

Example:

1. Import is started by scheduler at a given time
2. Importer checks once for the trigger file "trigger" if not available: fail
3. If trigger file is available "/home/import/trigger" and there is NO lock file "home/import/lock", create a lock file /home/import/lock" - otherwise fail
4. Copy data from /home/import/* to internal work folder
5. Remove trigger file
6. Remove lock file
7. Start importing data

Import from HTTP Endpoint

```
<httpInboundAdapter>
  <waitOnDownloadFailedInSeconds>5</waitOnDownloadFailedInSeconds>
  <retryCounter>0</retryCounter>
  <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  <httpResources>
    <httpResource>
      <resourceId>MultiChannelSalesQuery</resourceId>
      <resourceUrl>http://fiori-dev-hanadb01.gk.gk-
software.com:8000/MultiChannelSalesQuery</resourceUrl>
      <query><?format=xml</query>
      <batchSize>500</batchSize>
      <type>TransactionLog</type>
      <authentication>
        <user>xxxx</user>
        <encryptedPassword>xxxx</encryptedPassword>
      </authentication>
    </httpResource>
  </httpResources>
</httpInboundAdapter>
```

Code Block 4 Sample configuration

Tag	Mandatory Field	Value Range	Description
waitOnFileValidationFailedInSeconds	x	[0..Max Integer]	Number of seconds to wait when the connection attempt has failed. The call is restarted after that.
retryCounter	x	[0..Max Integer]	Number of repeat attempts when a connection attempt has failed. 0 means cancel after first error.
isArchiveToTaskEnabled	x	true, false	Backup of the input data in the archive. Data can be called later via the GUI.

HTTP Resource

Tag	Child Tag	Mandatory Field	Value Range	Default	Description
resourceId		x	[0..Max Integer]		ID of the resource. This is logged.
resourceUrl		x	[0..Max Integer]		URL of the resource, optionally with port. The query part can be indicated in the tag "query".
query			String	Empty	Is added at the end of the resourceURL.
httpMethod			POST, PUT, GET	GET	HTTP method
body			String		For httpMethod=POST and httpMethod=PUT, a body content may be entered here.
contentType			String		e.g. text/xml. If not set, no content type is set in the header.
socketTimeoutInSeconds			[0..Max Integer]	5	Maximum wait time for data after connection was established
connectionTimeoutInSeconds			[0..Max Integer]	5	Maximum time until connection is established with host
batchSize			[0..Max Integer]	0	Amount of data that can be loaded at once. Only for importers that support incremental loading. 0 means no incremental loading.
type		x	String		Type of the resource. The import can search this resource type to differentiate between several endpoints of various data streams.

Authentication

Credentials

If you use credential authentication (Digest, Basic), it's enough to specify user and encryptedPassword:

```
<httpInboundAdapter>
...
  <httpResource>
    ...
    <authentication>
      <user>xxxx</user>
      <encryptedPassword>xxxx</encryptedPassword>
    </authentication>
  </httpResource>
</httpResources>
</httpInboundAdapter>
```

Code Block 5 Sample configuration

Tag	Child Tag	Mandatory Field	Value Range	Default	Description
authentication	user	x	String		User
	encryptedPassword	x	String		Encrypted Password

The tag encryptedPassword supports an encrypted value. For more information see Import/Export Configuration: Credential Encryption.

OAuth2

```
<httpInboundAdapter>
...
  <httpResource>
    ...
    <authentication>
      <oauth2>
        <providerType>SAP</providerType>
        <tokenProviderUrl>https://csi-
gk.authentication.eu10.hana.ondemand.com/oauth/token</tokenProviderUrl>
        <clientId>sb-tlogcollector</clientId>
        <clientSecret>V0Bxxxxxx</clientSecret>
      </oauth2>
    </authentication>
  </httpResource>
</httpResources>
</httpInboundAdapter>
```

Code Block 6 Sample configuration

Tag	Child Tag	Mandatory Field	Value Range	Default	Description
authentication oauth2	providerType	x	SAP, CUSTOM		<ul style="list-style-type: none"> SAP: SAP OAuth2 Provider for accessing SAP Cloud systems. Sets grant type to "client_credentials" custom: Custom provider, concrete implementation is to be set later in plugin code.
	tokenProviderUrl	x	String		URL for requesting the access token
	clientId	x	String		Parameter client ID, passed to token provider
	clientSecret		String	Empty String	Parameter client Secret, passed to token provider

The tag clientSecret supports an encrypted value. For more information see Import/Export Configuration: Credential Encryption.

The flow is implemented this way:

1. Request an access token from the Token Provider and pass parameters to this call (POST)
2. Use this token for HTTP requests, set the token in the "Authorization" Header of the HTTP request

For custom provider implementations: You can add any custom child tags to <oauth2> and read the values of those tags in your custom authentication provider implementation.

SAP Passport

Tag	Child Tag	Mandatory Field	Value Range	Default	Description
sapPassport					Optional. Used for identification when accessing a SAP system.
	componentName	x	String		Describes the container/application instance.
	applicationComponentName	x	String		Component name of direct caller.
	tenantId	x	String		3-digit Number or empty String to identify a user.
	service	x	Integer		In Web AS: task type
	action	x	String		Action describes an activity in the Originator which triggered the Request.
	actionType	x	Integer		Depends on action

Import from/to S/FTP Endpoint

This adapter supports both FTP and SFTP connections. You need to configure the type and ports accordingly.

The example is for the inbound but also valid for the outbound (just replace the adapter from inbound to outbound).

```

<sftpInboundAdapter>
  <inboundType>SFTP</inboundType>
  <port>22</port>
  <host>192.168.0.2</host>
  <user>user</user>
  <encryptedPassword>password</encryptedPassword>
  <sftpConnectionTimeout>30</sftpConnectionTimeout>
  <downloadResources>
    <downloadResource>items.csv</downloadResource>
    <downloadResource>categories.csv</downloadResource>
  </downloadResources>
  <waitOnDownloadFailedInSeconds>5</waitOnDownloadFailedInSeconds>
</sftpInboundAdapter>

```

Code Block 7 Sample configuration

Tag	Mandatory Field	Value Range	Default	Description
inboundType	x	FTP, SFTP		FTP for accessing FTP server, SFTP for SFTP server
host	x	String		IP address of the SFTP server.
port	x	Integer		Port of the S/FTP server. Usually 22 for SFTP, port 21 for FTP
sftpConnectionTimeout		[0..Max Integer]	45	Timeout in Seconds of the SFTP connection.
waitOnFailedInSeconds	x	[0..Max Integer]		Number of seconds to wait when the connection attempt has failed. The call is restarted after that.
isArchiveToTaskEnabled		true,false	false	Backup of the input data in the archive. Data can be called later via the GUI.
retryCounter		[0..Max Integer]	5	Number of repeat attempts when a connection attempt has failed. 0 means cancel after first error.

downloadResources

Tag	Child Tag	Mandatory Field	Value Range	Description
downloadResources	downloadResource		String	Path to the file which will be imported. Regular expressions are supported.

Authentication

You can use either a password or a key to authentication.

Password Authentication

```
<sftpInboundAdapter>
  <user>user</user>
  <encryptedPassword>password</encryptedPassword>
</sftpInboundAdapter>
```

Code Block 8 Sample password authentication

Tag	Mandatory Field	Value Range	Description
user	x	String	Username for Authentication.
encryptedPassword	x	String	Encrypted Password for the user.

The tag encryptedPassword supports an encrypted value. For more information see Import/Export Configuration: Credential Encryption.

SSH Key Authentication

```
<sftpInboundAdapter>
  <user>user</user>
  <pathToKey>/run/secrets/key</pathToKey>
  <encryptedPassPhrase>passPhrase</encryptedPassPhrase>
</sftpInboundAdapter>
```

Code Block 9 Sample key authentication

Tag	Mandatory Field	Value Range	Default	Description
user	x	String		Username for Authentication.
pathToKey	x	String		Path to the key in the docker container. It is recommended to use a secret for the key. See details in the next section.
encryptedPassPhrase		String		Encrypted PassPhrase of the key.

Please see details about creating an providing a SSH key to the docker image in the Operation Guide, section "Using SSH Keys".

Using secret for pathToKey:

1. Configure a secret in your docker compose or k8s deployment file like this (docker compose example):
 secrets:
 - source: sftpsecret
 target: sftpsecrettarget
2. Then configure the pathToKey to use the target name of the secret in the path:
/run/secrets/sftpsecrettarget

If your SFTP access fails with: "...com.jcraft.jsch.JSchException: invalid privatekey" you are using an unsupported key format.
 Please check if you have created your key in the PEM format, e.g. like that: ssh-keygen -t rsa -m PEM
 You can use ssh-keygen to convert the key to the classic OpenSSH format:
 ssh-keygen -p -f file -m pem -P passphrase -N passphrase

The tag encryptedPassPhrase is supporting an encrypted value. For more information see Import/Export Configuration: Credential Encryption.

2.5.3 Import CSV Reading

2.5.3.1 Summary

This document describes the configuration details about reading CSV data.

2.5.3.2 Configuration of importing CSV Data

XML Structure

```
<csvReader>
  <charset>UTF-8</charset>
  <readHeader>true</readHeader>
  <escapeMode>EXCEL</escapeMode>
  <columnSeparator>|</columnSeparator>
  <textQualifier>"</textQualifier>
</csvReader>
```

csvReader

Tag	Optional	Default	Value Range	Description
charset	x	UTF-8	Charset Name	Charset used for reading data to the CSV
readHeader	x	true	true, false	Should the header be read
escapeMode	x	BACKSLASH	BACKSLASH EXCEL	Please see details below this table!
textQualifier	x	not set	Character	Please see details below this table!
columnSeparator	x		Character	Character to be used to separate cells

Details about Escaping

Escape Mode: Backslash

If you want to escape a string in a column, you have to use the backslash character: \ .

Imagine you use column separator | and you have this file:

```
pid|name
123|Michael | Jackson
```

To get the name field parsed, you need to escape the name column content using \.

```
pid|name
123|Michael \| Jackson
```

To escape \ you have to use doubled \\.

Escape Mode: Excel

For Excel escape mode you need to set the text qualifier as well! So imagine we set it this way:

```
<textQualifier>"</textQualifier>
```

Then we can escape the string like this:

```
pid|name
123|"Michael \| Jackson"
```

However, if " is also part of the text, you need to quote " by a double "":

Original:

```
pid|name
123|Michael " | Jackson
```

Quoted correctly:

```
pid|name
123|"Michael "" | Jackson"
```

2.5.4 Import XML Reading

2.5.4.1 Summary

This document describes the configuration details about reading XML data.

2.5.4.2 Configuration of importing XML Data

XML Structure

```
<xmlReader>
  <charset>UTF-8</charset>
</xmlReader>
```

xmlReader

Tag	Optional	Default	Value Range	Description
charset	x	UTF-8	Charset Name	Charset used for reading data to the file

2.5.5 Data Mapping by Configuration

Some imports support the mapping of CSV data fields to internal data fields and fixed values via XML configuration. Please see the concrete import documentation if this feature is supported.

Before version 2.1 we just supported column name mapping. In version 2.1 we introduced the new `<mode>` tag.

For the sake of backwards compatibility the default value for `<mode>` is `ColumnName` so old configurations are backwards compatible.

A supported import imports one or more entities. You have to use that entity for the configuration tag `<type>`.

2.5.5.1 Mapping by column name

Example: In your CSV file the price column is named "netUnitPrice", but the interface requires "regularUnitPrice" then you can add this mapping:

```
<fieldMappings>
  <fieldMapping>
    <type>Item</type>
    <mode>ColumnName</mode>
    <sourceField>netUnitPrice</sourceField>
    <destinationField>regularUnitPrice</destinationField>
  </fieldMapping>
</fieldMappings>
```

Besides mapping the column name you mark an optional field as mandatory this way:

```
<fieldMappings>
  <fieldMapping>
    <type>Item</type>
    <mode>ColumnName</mode>
    <sourceField>netUnitPrice</sourceField>
    <destinationField>regularUnitPrice</destinationField>
    <mandatory/>
  </fieldMapping>
</fieldMappings>
```

2.5.5.2 Mapping with a fixed value

Example: You would like to set a fixed value for all records for a given field:

```
<fieldMappings>
  <fieldMapping>
    <type>Item</type>
    <mode>FixedValue</mode>
    <fixedValue>12.99</fixedValue>
    <destinationField>regularUnitPrice</destinationField>
  </fieldMapping>
</fieldMappings>
```

2.5.5.3 Mark a field as mandatory

Example: if you want the field "regularUnitPrice" to be mandatory for the import, then you can add this mapping:

```
<fieldMappings>
  <fieldMapping>
    <type>Item</type>
    <mode>EnforceMandatory</mode>
    <destinationField>regularUnitPrice</destinationField>
  </fieldMapping>
</fieldMappings>
```

2.5.6 Import Timezones

2.5.6.1 Summary

Some importers can handle timezones. Importers that can handle timezones interpret the incoming date/time values with a configured timezone (request, configuration file). It depends on the importer if the time then is stored in UTC time zone or the business unit timezone.

Those importers document the timezone handling in their interface description.

If configuration is supported, there are several ways to configure the time zone to be used for interpreting the data of the import:

1. Pass the timezone in the import request
2. Configure a timezone in the configuration file

If you pass a timezone to the request, this one is used instead of anyone you've configured in the configuration file.

2.5.6.2 Timezone in the Request

You can pass a timezone using the parameter "**timezone**" when triggering your import:

```
http://...../?timezone=UTC
```

See here for more details about the format: <https://docs.oracle.com/javase/8/docs/api/java/time/ZonedDateTime.html#of-java.lang.String->

Please make sure your timezone is URL encoded, e.g. GMT+02:00 results in **GMT%2B02:00**

2.5.6.3 Timezone in the XML configuration

```
<import-config>
  <timeZone>GMT+02:00</timeZone>
  ...
</import-config>
```

The format resembles the one stated in the chapter about the request, except the URL encoding.

2.5.6.4 Timezone in the Business Unit

You can maintain a business unit specific timezone in the business unit's edit screen. Some importers might use that for storing the time in the database:

Globalization

Master currency
EUR☒ Activate rounding☒ Cash roundingScheduler time zone
GMT+01:00

2.5.7 Import Pipelines

2.5.7.1 Summary

With a pipeline for importers you can:

1. Run different importers in a sequence
2. Define one single endpoint for an importer that accepts data for a couple of business units and dispatches them to the receiver business unit(s)

To run more different importers in sequence, you just have to add them to a new pipeline using the GUI. For the second case you need additional configuration in your importer XML configuration.

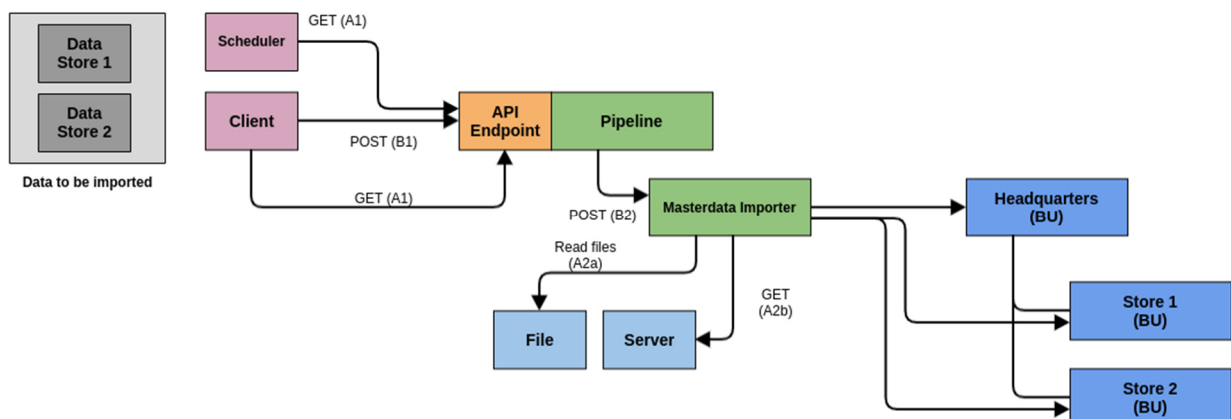
In a pipeline, the importer supports two modes:

1. Push: Data is pushed via HTTP POST to the orchestrator that dispatches the data to the business units
2. Pull: The orchestrator triggers the business units to get data via file or HTTP

For the push case there are two different ways for the importer to find out which business units should get the data (= dispatching):

1. Analyze the file names of the input data
2. Analyze the content of the input data

2.5.7.2 Overview



Triggering the Pipeline

Pushing the Data

PUSH: Post data to the API endpoint. Push can be done via HTTP POST or the drag and drop in the GUI.

1. B1: POST a zip/gzip container or a single file to the API endpoint
2. B2: Data is passed to the master data importer
3. The master data importer then analyzes the data depending on the configured strategy (e.g. by file name or checking the content of the file)
4. The dispatcher then extracts the data for each business unit
5. Each business unit receives her data and starts importing data in an own task
6. The task is processed on any node (processing or controller node)
7. The pipeline waits until all tasks are processed and returns with the results and protocols

Pulling the Data

Pulling can be triggered either by the GUI or the scheduler or a HTTP GET call.

PULL: Trigger the pipeline by sending a GET

1. A1: Send a GET to the API endpoint by a client or by configuring a scheduler in the pricing UI
2. A2a: Read the data from the filesystem (or mount point, e.g. SMB) or A2b: Download data via HTTP GET from a server. For both cases you can use the business unit id to adapt the path in the filesystem or URL depending on each business unit that is triggered.
3. The master data importer then analyzes the data depending on the configured strategy (e.g. by file name or checking the content of the file)
4. The dispatcher then extracts the data for each business unit
5. Each business unit receives her data and starts importing data in an own task
6. The task is processed on any node (processing or controller node)
7. The pipeline waits until all tasks are processed and returns with the results and protocols

2.5.7.3 Scenarios

Currently we do not support linking more importers to the pipeline and send one data package to the pipeline API endpoint to dispatch the input data to the appropriate importer.

For this case we recommend to create importers and use their API endpoints.

Push Scenarios

You are using many business units - e.g. for computing prices on the store level - and need to send data to all of them. Then the pipeline supports you to:

- Send a data package for each business unit using the pipeline API endpoint. The endpoint then dispatches the package to the appropriate business unit.
- Send a data package to the pipeline API endpoint that the dispatcher should send to all business units in the hierarchy
- Send a large data package containing data for several business units at once to one API endpoint. The dispatcher then checks each file and finally creates new (smaller) packages that are send to the corresponding business unit

Send a data package to the pipeline API endpoint to send it to all business units in the hierarchy

Configuration

- Create a business unit hierarchy
- Create an importer in the top level business unit and enable the master hierarchy flag
- In the importer's XML configuration add an empty "orchestration" section
- Create a pipeline and link the importer
- Post your package to the pipeline API

API Change: In older versions before 3.2.0 this behavior was triggered by removing the orchestration section. We changed that to be able to support the new use case for excluding the top level business unit also for push.

```
<orchestration>
</orchestration>
```

Send a data package to the pipeline API endpoint to send it to all business units in the hierarchy except the top level business unit

Configuration

- Create a business unit hierarchy
- Create an importer in the top level business unit and enable the master hierarchy flag
- In the importer's XML configuration add a "orchestration" section and setting the exclude parameter to true
- Create a pipeline and link the importer
- Post your package to the pipeline API

Tag	Mandatory	Default	Value Range	Description	Example
isExcludeTopLevelNodeForPushEnabled		false	true, false	true: the root business unit in the hierarchy is not triggered but all other sub business units in the hierarchy false: all business units in the hierarchy are triggered	true

```
<orchestration>
  <isExcludeTopLevelNodeForPushEnabled>true</isExcludeTopLevelNodeForPushEnabled>
</orchestration>
```

Dispatching Data based on File Names of the Input Data

Dispatching data based on file names can be configured for two different levels:

1. Analyze the zip/gzip container file name or the file name if it is not a zip/gzip container.
2. Analyze the name of the files in the container

```

<orchestration>
  <fileNameDispatcher>
    <isDispatchByContainerFileNameEnabled>true</isDispatchByContainerFileNameEnabled>
    <isDispatchByFileNamesInContainerEnabled>true</isDispatchByFileNamesInContainerEnabled>
    <businessUnitMatcher>^[0-9]+</businessUnitMatcher>
    <businessUnitTemplate>store{0}</businessUnitTemplate>
  </fileNameDispatcher>
</orchestration>

```

Tag	Default	Value Range	Description	Example
isDispatchByContainerFileNameEnabled		true, false	If true, analyze the file name of the zip/gzip container or a single file if no container is used. Checks if the businessUnitMatcher regular expression matches to extract the business unit ID from the file name.	true
isDispatchByFileNamesInContainerEnabled		true, false	If true, analyze the file name of file(s) in the zip/gzip container. Checks if the businessUnitMatcher regular expression matches to extract the business unit ID from a file name. Runs through all files. Stops if a file name matched.	true
businessUnitMatcher		Regular expression	Extract the business unit id from a file name.	^[0-9]+
businessUnitTemplate		String with or without {0}	If an ID is extracted then you can use that template to define how the final business unit ID is constructed. {0} is the placeholder for the extracted ID from the file name.	store{0}

Analyze the zip/gzip container file name or the file name

If this strategy is enabled (parameter: isDispatchByContainerFileNameEnabled), then either the name of a gzip/zip container is analyzed or the name of a single file if no container is used.

The dispatcher applies the businessUnitMatcher regular expression to the file name (container or single file). If the matcher is successful, then an ID is extracted.

If you also enabled isDispatchByFileNamesInContainerEnabled to look into a zip/gzip container, then this strategy is not executed if this strategy already extracted an ID. Only if this strategy failed.

To construct the final business unit ID that is used to address the corresponding business unit, you define a template like "store{0}". This means that the prefix "store" is added to the extracted ID from the file name.

This final business unit is then used for dispatching the content to a business unit.

If we talk about a business unit ID we **DO NOT** mean the name of a business unit. If you create a business unit you can also provide a business unit ID besides the name. This is optional. If you don't define it, the system creates a unique ID for you.

However, auto-generated IDs won't help you here if you plan to use the pipeline with the dispatcher. So it's important for you to define a reasonable ID that fits to your file names.

To be able to analyze file names that are not in zip/gzip containers, we expect for POST HTTP calls a properly set HTTP Header for the file name, e.g.:

Content-Disposition: filename=1_items.zip

Example: Send multiple ZIP files, each with data for one business unit

The example case is:

- Send one ZIP container with items for one store named 1_items.zip to the pipeline API endpoint
- Import the data into the one business unit store1

What you have to do:

1. You create a master data CSV importer and enable the flag "Master for business unit hierarchy".
2. In the XML configuration of the master data CSV importer the orchestration is configured this way:

```
<orchestration>
  <fileNameDispatcher>
    <isDispatchByContainerFileNameEnabled>true</isDispatchByContainerFileNameEnabled>

    <isDispatchByFileNamesInContainerEnabled>false</isDispatchByFileNamesInContainerEnabled>
    <businessUnitMatcher>^[0-9]+</businessUnitMatcher>
    <businessUnitTemplate>store{0}</businessUnitTemplate>
  </fileNameDispatcher>
</orchestration>
```

3. You create a pipeline and link this importer and confirm
4. The endpoint API then is displayed in the details panel
5. You have a GZIP container 1_items.zip with one item file: 1_items.csv
6. POST the GZIP file to the endpoint (**and don't forget to set the content-disposition header!**)

What happens step by step:

- The dispatcher receives the file 1_items.zip. A file name dispatcher is configured. It should check for the container's name (isDispatchByContainerFileNameEnabled=true)
- Analyze the container's file name: 1_items.zip.
- The regular expression pattern is applied and returns an ID: "1"
- Then the template is used to construct the business unit id: store1.
- The file 1_items.zip is passed to a newly created task for the master data importer for business unit store1.
- After the importer is done, the pipeline finishes and returns to the caller.

Analyze the name of the files in the container

If this strategy is enabled (parameter: isDispatchByFileNamesInContainerEnabled), then the name(s) of files in the gzip/zip container are analyzed.

The dispatcher applies the businessUnitMatcher regular expression to each file name in the container to determine the receiver (= business unit). If the matcher is successful, then an ID is extracted.

To construct the final business unit ID that is used to address the corresponding business unit, you define a template like "store{0}". This means that the prefix "store" is added to the extracted ID from the file name.

This final business unit is then used for dispatching the content to a business unit. For each business unit, a separate task with an importer for this business unit is started.

If we talk about a business unit ID we **DO NOT** mean the name of a business unit. If you create a business unit you can also provide a business unit ID besides the name. This is optional. If you don't define it, the system creates a unique ID for you.

However, auto-generated IDs won't help you here if you plan to use the pipeline with the dispatcher. So it's important for you to define a reasonable ID that fits to your file names.

Example: Send a ZIP file with data for multiple business units

The example case is:

- Send one ZIP container with items for two stores named items.zip to the pipeline API endpoint
- Import the data into the two business units store1 and store2

What you have to do:

1. You create a master data CSV importer and enable the flag "Master for business unit hierarchy".

2. In the XML configuration of the master data CSV importer the orchestration is configured this way:

```
<orchestration>
  <fileNameDispatcher>
    <isDispatchByContainerFileNameEnabled>false</isDispatchByContainerFileNameEnabled>
    <isDispatchByFileNamesInContainerEnabled>true</isDispatchByFileNamesInContainerEnabled>
    <businessUnitMatcher>^[0-9]+</businessUnitMatcher>
    <businessUnitTemplate>store{0}</businessUnitTemplate>
  </fileNameDispatcher>
</orchestration>
```

3. You create a pipeline and link this importer and confirm

4. The endpoint API then is displayed in the details panel

5. You have a GZIP container items.zip with two item files: 1_items.csv, 2_items.csv

6. POST the GZIP file to the endpoint

What happens step by step:

- The dispatcher receives the file items.zip. A file name dispatcher is configured. It should not check for the container's name (isDispatchByContainerFileNameEnabled = false), but to check the files in the zip file one by one (isDispatchByFileNamesInContainerEnabled=true)
- Decompress the zip file
- Analyze the first file: 1_items.csv.
- The regular expression pattern is applied and returns an ID: "1"
- Then the template is used to construct the business unit id: store1.
- The file 1_items.csv is passed to a newly created task for the master data importer for business unit store1.
- Analyze the second file: 2_items.csv.
- The regular expression pattern is applied and returns an ID: "2"
- Then the template is used to construct the business unit id: store2.
- The file 2_items.csv is passed to a newly created task for the master data importer for business unit store2.
- After all importers are done, the pipeline finishes and returns to the caller

Dispatching Data based on Content of the Input Data

Dispatching CSV Content

You can configure the dispatcher to open the CSV file and check the content of a column. E.g. if the column with name "store" contains the ID, then this can be extracted from the first record of the file.

To construct the final business unit ID that is used to address the corresponding business unit, you define a template like "store{0}". This means that the prefix "store" is added to the extracted ID from the CSV content.

This final business unit is then used for dispatching the content to a business unit. For each business unit, a separate task with an importer for this business unit is started.

If we talk about a business unit ID we **DO NOT** mean the name of a business unit. If you create a business unit you can also provide a business unit ID besides the name. This is optional. If you don't define it, the system creates a unique ID for you.

However, auto-generated IDs won't help you here if you plan to use the pipeline with the dispatcher. So it's important for you to define a reasonable ID that fits to your CSV content.

Configuration

Tag	Mandatory	Value Range	Description	Example
businessUnitColumnName	either this or businessUnitColumnIndex	String	If true, analyze the file name of the zip/gzip container or a single file if no container is used. Checks if the businessUnitMatcher regular expression matches to extract the business unit ID from the file name.	store
businessUnitColumnIndex	either this or businessUnitColumnName	Integer [0..]		1
businessUnitTemplate	x	String with or without {0}	If an ID is extracted then you can use that template to define how the final business unit ID is constructed. {0} is the placeholder for the extracted ID from the file CSV content.	store{0}

Example for dispatching by column name:

```
<orchestration>
  <csvContentDispatcher>
    <businessUnitColumnName>storeID</businessUnitColumnName>
    <businessUnitTemplate>store{0}</businessUnitTemplate>
  </csvContentDispatcher>
</orchestration>
<csvReader>
  <charset>UTF-8</charset>
  <readHeader>true</readHeader>
  <escapeMode>EXCEL</escapeMode>
  <columnSeparator>|</columnSeparator>
  <textQualifier>"</textQualifier>
</csvReader>
```

Please mind that you need to configure the CSV reading for CSV content dispatching too. Internally the dispatcher uses the CSV reading subsystem. If you forget to configure that, the dispatcher fails. See here for more details.

Example for dispatching by column index:

```
<orchestration>
  <csvContentDispatcher>
    <businessUnitColumnIndex>0</businessUnitColumnIndex>
    <businessUnitTemplate>store{0}</businessUnitTemplate>
  </csvContentDispatcher>
</orchestration>
<csvReader>
  <charset>UTF-8</charset>
  <readHeader>true</readHeader>
  <escapeMode>EXCEL</escapeMode>
  <columnSeparator>|</columnSeparator>
  <textQualifier>"</textQualifier>
</csvReader>
```

Dispatching XML Content

You can configure the dispatcher to open the XML file and check the content of a column. E.g. if the column with name "store" contains the ID, then this can be extracted from the first record of the file.

To construct the final business unit ID that is used to address the corresponding business unit, you define a template like "store{0}". This means that the prefix "store" is added to the extracted ID from the XML content.

This final business unit is then used for dispatching the content to a business unit. For each business unit, a separate task with an importer for this business unit is started.

If we talk about a business unit ID we **DO NOT** mean the name of a business unit. If you create a business unit you can also provide a business unit ID besides the name. This is optional. If you don't define it, the system creates a unique ID for you.

However, auto-generated IDs won't help you here if you plan to use the pipeline with the dispatcher. So it's important for you to define a reasonable ID that fits to your XML content.

Configuration

Tag	Mandatory	Value Range	Description	Example
businessUnitXmlPath	x	String with . as XML tag separator	Define a XML path to a tag. The data of this tag is extracted and used as an ID that is put to the businessUnitTemplate as input. If this is a tag hierarchy, use "." to separate the tags.	root.header.store
businessUnitTemplate	x	String with or without {0}	If an ID is extracted then you can use that template to define how the final business unit ID is constructed. {0} is the placeholder for the extracted ID from the file CSV content.	store{0}

Example for dispatching by the XML tag <root><header><store>:

```
<orchestration>
  <xmlContentDispatcher>
    <businessUnitXmlPath>root.header.store</businessUnitXmlPath>
    <businessUnitTemplate>store{0}</businessUnitTemplate>
  </xmlContentDispatcher>
</orchestration>
<xmlReader>
  <charset>UTF-8</charset>
</xmlReader>
```

Please mind that you need to configure the XML reading for XML content dispatching too. Internally the dispatcher uses the XML reading subsystem. If you forget to configure that, the dispatcher fails.
See here for more details.

Pull Scenarios

- Trigger the pipeline by a scheduler that triggers all business units in the hierarchy to download data via HTTP GET
- Trigger the pipeline by a scheduler that triggers all business units (except the root business unit) in the hierarchy to download data from a filesystem

Trigger the pipeline by a scheduler that triggers all business units in the hierarchy to download data via HTTP GET

Configuration

```
<orchestration>
  <isExcludeTopLevelNodeForPullEnabled>>false</isExcludeTopLevelNodeForPullEnabled>
</orchestration>
```

Tag	Mandatory	Default	Value Range	Description	Example
isExcludeTopLevelNodeForPullEnabled		false	true, false	true: the root business unit in the hierarchy is not triggered but all other sub business units in the hierarchy false: all business units in the hierarchy are triggered	true

Besides that you can configure a variable resource URL depending on the business unit in your httpInboundAdapter configuration. There is a variable BU that contains the business unit ID. You can use that in your resourceUrl tag to make the call business unit specific, e.g. as part of the path or as parameter.

As this parameter is optional, you don't need to add it to your configuration if you need the default (false).

Tag	Mandatory	Default	Value Range	Description
resourceUrl	x		String	The URL that should be called to download the data. Supported variables: <ul style="list-style-type: none"> • \${BU} Contains the business unit id of the triggered business unit

Example

```
...
<orchestration>
  <isExcludeTopLevelNodeForPullEnabled>>false</isExcludeTopLevelNodeForPullEnabled>
</orchestration>
<httpInboundAdapter>
  <waitOnDownloadFailedInSeconds>5</waitOnDownloadFailedInSeconds>
  <retryCounter>0</retryCounter>
  <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  <httpResources>
    <httpResource>
      <resourceId>TransactionLog</resourceId>
      <resourceUrl>http://myServer.de/TransactionLog?store=${BU}</resourceUrl>
      <query></query>
      <batchSize>0</batchSize>
      <type>TransactionLog</type>
    </httpResource>
  </httpResources>
</httpInboundAdapter>
```

Trigger the pipeline by a scheduler that triggers all business units (except the root business unit) in the hierarchy to download data from a filesystem

Configuration

```
<orchestration>
  <isExcludeTopLevelNodeForPullEnabled>true</isExcludeTopLevelNodeForPullEnabled>
</orchestration>
```

Tag	Mandatory	Default	Value Range	Description	Example
isExcludeTopLevelNodeForPullEnabled		false	true, false	true: the root business unit in the hierarchy is not triggered but all other sub business units in the hierarchy false: all business units in the hierarchy are triggered	true

Besides that you can configure a variable import folder depending on the business unit in your httpInboundAdapter configuration. There is a variable BU that contains the business unit ID. You can use that in your importDir tag to make the call business unit specific as part of the path.

■ As this parameter is optional, you don't need to add it to your configuration if you need the default (false).

Tag	Mandatory	Default	Value Range	Description
importDir	x		String	The folder from which to import data for this business unit. Supported variables: <ul style="list-style-type: none">• \${BU} Contains the business unit id of the triggered business unit

Example

```
...
<orchestration>
  <isExcludeTopLevelNodeForPullEnabled>true</isExcludeTopLevelNodeForPullEnabled>
</orchestration>
<fileInboundAdapter>
  <importDir>/import/${BU}</importDir>
  <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  <copyOption>NONE</copyOption>
  <isLockFileEnabled>false</isLockFileEnabled>
  <lockFileName>lock</lockFileName>
  <triggerOption>NO_TRIGGER</triggerOption>
  <triggerFileName>trigger</triggerFileName>
  <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
  <fileValidationRetryCount>0</fileValidationRetryCount>
</fileInboundAdapter>
```


2.5.7.4 Additional Configuration Options using Business Unit Parameters

Besides the configuration options mentioned above, the configuration file also supports business unit parameters that are injected at any tag of your configuration file.

For details, please see the section "Import/Export Configuration: Business Unit Parameters" in the Configuration Guide.

2.5.8 Competitor Prices Import

2.5.8.1 Competitor Prices Import - CSV

Summary

Import of competitors and competitor prices in **CSV Format** as CSV file or via an HTTP endpoint.

Modes

Mode	Default	Description
INITIAL	x	Delete current competitor prices for all competitors. Do not delete competitors if they exist. You can delete them using the GUI. However, this also deletes the historical competitor prices.
UPDATE		Update competitor prices and competitors

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET		<ul style="list-style-type: none">Ad hoc reading of one or more files from the file system.Ad hoc querying of a ZIP file from an endpoint via HTTP GET.
POST		<ol style="list-style-type: none">A file is transmitted.One or more files in the ZIZ archive are transmitted.
Scheduler		<ul style="list-style-type: none">Time-controlled reading of one or more files from the file system.Time-controlled querying of a ZIP file from an endpoint via HTTP GET.

Mapping and Validation

Competitors

Field	Format	Example	Mandatory field	Description	Validation and Error Code
competitorId	String	Amazon	x	Competitor ID	<ul style="list-style-type: none">not set: 51011
description	String			Description	

Example

The file name must match the regular expression in the Configuration file .

```
competitorId|description
Amazon|Amazon Europa
Zalando|Zalando.de
```

Code Block 10 competitors.csv

Competitor Prices

Field	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code
itemId	String	33838	x	Item ID		<ul style="list-style-type: none">not set: 51011
competitorId	String	Amazon	x	Reference to the corresponding competitor ID		<ul style="list-style-type: none">not set: 51011
price	Decimal number	5.99	x	Competitor prices	"," as separator of decimal places	<ul style="list-style-type: none">not set: 51011value is invalid or < 0: 51016

Example

```
itemId|competitorId|price  
id0815|Amazon|1.99  
id0815|Zalando|2.05
```

Code Block 11 competitorprices.csv

Configuration

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

CSV Reading

Please see details about the configuration [here](#).

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>Competitor</mappingId>
      <regExpression>competitors.*csv</regExpression>
      <type>Competitor</type>
      <defaultName>competitors.csv</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>CompetitorPrice</mappingId>
      <regExpression>competitorprice.*csv</regExpression>
      <type>CompetitorPrice</type>
      <defaultName>competitorprice.csv</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <csvReader>
    <charset>UTF-8</charset>
    <readHeader>true</readHeader>
    <escapeMode>BACKSLASH</escapeMode>
    <columnSeparator>|</columnSeparator>
  </csvReader>
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

Code Block 12 Sample configuration

2.5.9 Transaction Log Import

2.5.9.1 Transaction Log Import - CSV

Summary

This import is used to import transaction logs in CSV format from the file(s) or from an HTTP endpoint. Transaction logs are, for example, transactions from stationary POS systems or from an e-commerce system.

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET		<ul style="list-style-type: none">Ad hoc reading of one or more files from the file system.Ad hoc querying of data from an endpoint via HTTP GET.
POST		<ul style="list-style-type: none">One single file or multiple files in a ZIP archive are transmitted via HTTP POST.
PUT		<ul style="list-style-type: none">One single file or multiple files in a ZIP archive are transmitted via HTTP PUT.
Scheduler		<ul style="list-style-type: none">Time-controlled reading of one or more files from the file system.Time-controlled querying of a file from an endpoint via HTTP GET, PUT or POST.

Data

A transaction represents an event. In the case of purchase orders, a transaction refers to a line item of the order. If several line items exist, these must be distributed with the same transaction ID in several transaction entries.

The format is compatible with RDE Server Translog CSV format.

Example

```
time|transactID|groupID|itemID|transType|basket|order|itemsAction|categoryPath|price|channelID
2017-09-01 10:05:37|101700903820|0|747425|1|0|1||25.83|store
2017-09-01 10:04:33|101700903837|0|526784|1|0|1||22.17|store
2017-09-01 10:12:51|101700904056|0|5012|1|0|1||70.88|store
2017-09-01 10:12:51|101700904056|0|13154|1|0|3||17.76|store
2017-09-01 10:12:51|101700904056|0|250320|1|0|1||33.20|store
```

This example contains fields that are not supported by the import like itemsAction and categoryPath. These fields are typically sent by an RDE server and are discarded.

CSV field	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code															
time	YYYY-MM-DD hh:mm:ss or YYYY-MM-DDThh:mm:ss	2018-01-05 15:45:33 2018-01-05T15:45:33	x	Timestamp of the transaction without timezone . Please see next chapter about handling timezones.	Convert to timestamp (long value in seconds)	<ul style="list-style-type: none">not set: 51011															
transactID	String	928888281	x	Transaction ID. When transmitting several line items of a purchase order, an entry must be sent for each line item. The transaction ID may be the same for all line items.		<ul style="list-style-type: none">not set: 51011															
itemID	String	1245		Item ID		<ul style="list-style-type: none">not set: 51011															
transType	-1, -2, 0, 1	1	x	Transaction type. In addition to purchase orders, tracking transactions from the e-commerce system, too, such as search/click/shopping cart, can be used. Whether this information is used depends on the particular pricing strategy.	<table><tr><th>Value</th><th>Mapping</th><th>Meaning</th></tr><tr><td>-2</td><td>s</td><td>search</td></tr><tr><td>-1</td><td>c</td><td>Click</td></tr><tr><td>0</td><td>b</td><td>shopping cart</td></tr><tr><td>1</td><td>o</td><td>purchase order</td></tr></table>	Value	Mapping	Meaning	-2	s	search	-1	c	Click	0	b	shopping cart	1	o	purchase order	<ul style="list-style-type: none">not set: 51011invalid value: 51017
Value	Mapping	Meaning																			
-2	s	search																			
-1	c	Click																			
0	b	shopping cart																			
1	o	purchase order																			
basket	integer [1..max integer], if transType = 0, otherwise 0 or empty		x for transType 0 (shopping cart)	Number of pieces of the item that was placed in the shopping cart.	Evaluated if transType = 0	<ul style="list-style-type: none">not set: 51011if transType is basket:value < 1: 51018value is not a valid number: 51014else:invalid value: 51017															
order	integer [1..max integer], if transType = 1, otherwise 0 or empty		x for transType 1 (order)	Number of pieces of the item that was placed in the shopping cart.	Evaluated if transType = 1	<ul style="list-style-type: none">not set: 51011if transType is order:value < 1: 51018value is not a valid number: 51014else:invalid value: 51017															
price	Floating point with "." as separator of decimal places. No thousands, maximum two decimal places.	15.99	x for transtype 0,1 (order, shopping cart)	Line item price (gross): Item price (not multiplied by quantity!). Assigned discounts are already deducted.	Evaluated if transType = 0 or = 1	<ul style="list-style-type: none">not set: 51011value is not a valid price or value < 0: 51016															

netPrice	Floating point with "." as separator of decimal places. No thousands, maximum two decimal places.	14.99		Line item price (net): Item price (not multiplied by quantity!). Assigned discounts are already deducted.	Evaluated if transType = 0 or = 1 If the value is not set, then the value of the field "price" is used.	<ul style="list-style-type: none"> not set: 51011 value is not a valid price or value < 0: 51016
promotionAmount	Floating point with "." as separator of decimal places. No thousands, maximum two decimal places.	1.23		Sum of line item discounts or total discounts granted for this line item, for one item (i.e. without considering the quantity).	Evaluated if transType = 0 or = 1	<ul style="list-style-type: none"> not set: 51011 value is not a valid price or value < 0: 51016
channelID	String	online		Name or ID of the sales channel. Is used, for example, to differentiate between offline/online channels.		

Handling Timezones

The "time" column does not contain any timezone. However, you can tell the importer to interpret the time in a given timezone in this priority:

1. Pass the timezone parameter to the POST/GET request
2. Configure a timezone in the XML configuration
3. **Default: UTC timezone**

Then the importer interprets the time in the time column using this timezone and eventually converts this time to the timezone of the business unit if the business unit's timezone is different.

See more details in the chapter "Import Timezones".

Example:

```
time|transactID|groupID|itemID|transType|basket|order|itemsAction|categoryPath|price|channelID
2017-09-01T10:05:37|101700903820|0|747425|1|0|1|||25.83|store
```

Let's imagine you configured the timezone GMT+02:00 in the business unit. The importer treats the time 2017-09-01 10:05:37 as UTC and converts it to the GMT+02:00 timezone of the business unit: 2017-09-01 12:05:37.

If you configure in the importer configuration: `<timeZone>GMT+02:00</timeZone>`
for 2017-09-01 10:05:37 the timezone GMT+2 is used and no conversion takes place as the business unit is also GMT+02:00: 2017-09-01 10:05:37.

Configuration

Tag	Format	Mandatory Field	Default	Description
importOrdersEnabled	true, false		true	Import order transactions?
importBasketsEnabled	true, false		false	Import basket transactions?
importSearchesEnabled	true, false		false	Import search transactions?
importClicksEnabled	true, false		false	Import click transactions?
importMode	INITIAL, UPDATE		UPDATE	INITIAL = Deletes transaction data before the import starts. UPDATE = Updates existing entities and adds new entities.
timeZone	See chapter "Import Timezones". E.g. GMT+02:00		not set	Timezone to use for the time that is passed in column "time".

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

For initial import of many files in one zip container, please see the protocol setting for aggregating the statistics to one value here.

More than 1000 files per zip file can harm lead to very large statistics leading to memory problems.

Import via File System

File Names

File naming follows the following rules:

- regular expression: translog.*\csv
- translog.yyyyMMdd

These can be freely defined in the configuration. See section "Configuration".

Sorting

If there is a timestamp at the end of the file names, e.g. translog,20180102, the files are sorted and evaluated in ascending order of the date. If two files have the same timestamp, the order is undefined. If the file names do not have a timestamp, the files are sorted alphabetically in ascending order.

Configuration

See "Import Communication Interface".

Import via HTTP Endpoint

Configuration

See "Import Communication Interface".

CSV Reading

Please see details about the configuration here.

Data Mapping

This import supports mapping of CSV data fields to internal data fields and fixed values via XML configuration.

This import supports mapping of fields of those entities:

Entity	<type>
Transactions	TransactionLog

For more details about the mapping, please refer to the documentation here.

Mapping example

```
<fieldMappings>
  <fieldMapping>
    <type>TransactionLog</type>
    <mode>ColumnName</mode>
    <sourceField>timeStamp</sourceField>
    <destinationField>time</destinationField>
  </fieldMapping>
</fieldMappings>
```

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <importOrdersEnabled>true</importOrdersEnabled>
  <importSearchesEnabled>false</importSearchesEnabled>
  <importBasketsEnabled>false</importBasketsEnabled>
  <importClicksEnabled>false</importClicksEnabled>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>translogWithoutDate</mappingId>
      <regExpression>translog.*csv</regExpression>
      <type>TransactionLog</type>
      <defaultName>translog.csv</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>translogWithDate</mappingId>
      <regExpression>translog\..*</regExpression>
      <type>TransactionLog</type>
      <defaultName>translog.csv</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <sorting>com.prudsys.plugins.pricing.pricingengine.importservice.transactionlog.transactionlogcsv.TransactionLogCsvFileComparator</sorting>
</import-config>
</inboundDataMappings>
<csvReader>
  <charset>UTF-8</charset>
  <readHeader>true</readHeader>
  <escapeMode>BACKSLASH</escapeMode>
  <columnSeparator>|</columnSeparator>
</csvReader>
<fileInboundAdapter>
  <importDir>import</importDir>
  <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  <copyOption>NONE</copyOption>
  <isLockFileEnabled>false</isLockFileEnabled>
  <lockFileName>lock</lockFileName>
  <triggerOption>NO_TRIGGER</triggerOption>
  <triggerFileName>trigger</triggerFileName>
  <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
  <fileValidationRetryCount>0</fileValidationRetryCount>
</fileInboundAdapter>
</import-config>
```

2.5.9.2 Transaction Log Import - POSLog

Summary

This import is used to import transaction logs in POSLog 3.0 XML format from the file(s) or from an HTTP endpoint. Transaction logs are, for example, transactions from stationary POS systems or from an e-commerce system.

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET		<ul style="list-style-type: none">• Ad hoc reading of one or more files from the file system.• Ad hoc querying of data from an endpoint via HTTP GET.
POST		<ul style="list-style-type: none">• One single file or multiple files in a ZIP archive are transmitted via HTTP POST.
Scheduler		<ul style="list-style-type: none">• Time-controlled reading of one or more files from the file system.• Time-controlled querying of a file from an endpoint via HTTP GET, PUT or POST.

POSLog Tag	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code
<Transaction>			x	All actions that can be recorded by the cash register	Transactions with <ul style="list-style-type: none"> MajorVersion ≠ "3", MinorVersion ≠ "0", CancelFlag ≠ "false" or TrainingModeFlag ≠ "false" are not imported.	
<Transaction> <RetailTransaction> <LineItem>			x	The actual transaction according to our translogs	LineItems with VoidFlag ≠ "false" are not imported.	
<Transaction> <RetailTransaction> <LineItem> <Sale>			x		LineItems are not imported if the attribute ItemType in the sale tag is not set to "Stock" or "SES:Pickup"	
<Transaction> <RetailTransaction> <LineItem> <BeginDateTime>	String yyyy-MM-dd'T'HH:mm:ss.SS XXXX	2018-07-31T17:25:34.000+02:00	x	Timestamp of the transaction		
<Transaction> <RetailTransaction> <LineItem> <Sale> <ItemID>	String	31100224	x	Item ID		<ul style="list-style-type: none"> not set: 510 11
<Transaction> <RetailTransaction> <LineItem> <Sale> <Quantity>	Floating point with "." as separator of decimal places.	1.000	x	Quantity of sold units.	If the attribute UnitOfMeasureCode ≠ "PCE", the quantity is statically fixed at the value 1. Entries where quantity < 1 are not imported.	<ul style="list-style-type: none"> not set: 510 11 value < 1: 510 31 invalid value: 510 30
<Transaction> <InternalTransactionID>	String	55077e80f9acb9149b2a57d3effd2740	x	Constitutes, together with, <Transaction> <RetailTransaction> <LineItem> <SequenceNumber> the transaction ID, according to the following pattern: InternalTransactionID-SequenceNumber This ID is used in all underlying LineItems.		<ul style="list-style-type: none"> not set: 510 11

<Transaction> <RetailTransaction> <LineItem> <SequenceNumber> >	String	0	x	Constitutes, together with, <Transaction> <InternalTransactionID> the transaction ID, according to the following pattern: InternalTransactionID-SequenceNumber		<ul style="list-style-type: none"> not set: 51011
<Transaction> <RetailTransaction> <LineItem> <Sale> <ExtendedAmount> >	Decimal number with "." as separator of decimal places	2975.00	x	The price is a result of the quotient from the ExtendedAmount over Quantity; The price is rounded to two decimal places		<ul style="list-style-type: none"> not set: 51011 value is invalid or value < 0: 51016
<Transaction> <RetailTransaction> <LineItem> <Sale> <RegularSalesUnitPrice>	Decimal number with "." as separator of decimal places	2975.00		The discount is a result of the difference from RegularSalesUnitPrice minus price (see ExtendedAmount); The discount is rounded to two decimal places		
<Transaction> <RetailTransaction> <LineItem> <Sale> <Tax> <Amount>	Decimal number with "." as separator of decimal places	7.7820		If set, this is subtracted from the extended amount in order to import only net prices		
<Transaction> <RetailTransaction> <LineItem> <Sale> <POSIdentity> <POSItemID>	String	1234	x	The item ID		<ul style="list-style-type: none"> not set: 51011

Handling Timezones

The importer uses the timezone that is passed in the <BeginDateTime> tag to interpret the time, converts it to the business unit timezone and saves it to the.

Example:

```
<BeginDateTime>2018-07-31T17:25:34.000+02:00</BeginDateTime>
Business Unit Timezone: GMT+02:00
Is saved as 2018-07-31 17:25:34.000.
```

Configuration

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

For initial import of many files in one zip container, please see the protocol setting for aggregating the statistics to one value here.

More than 1000 files per zip file can harm lead to very large statistics leading to memory problems.

General Settings

Tag	Format	Mandatory Field	Default	Description
importMode	INITIAL, UPDATE		UPDATE	INITIAL = Deletes transaction data before the import starts. UPDATE = Updates existing entities and adds new entities.

Import via File System

File Names

File naming follows the following rules:

- regular expression: translog.**.csv
- translog.yyyyMMdd

These can be freely defined in the configuration. See section "Configuration".

Sorting

If there is a timestamp at the end of the file names, e.g. translog,20180102, the files are sorted and evaluated in ascending order of the date. If two files have the same timestamp, the order is undefined. If the file names do not have a timestamp, the files are sorted alphabetically in ascending order.

Configuration

See "Import Communication Interface".

Import via HTTP Endpoint

Configuration

See "Import Communication Interface".

XML Reading

Please see details about the configuration here.

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <channelID>Store</channelID>
  <xmlReader>
    <charset>UTF-8</charset>
  </xmlReader>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>TransactionLog</mappingId>
      <regExpression>SellingTransactions-.*</regExpression>
      <type>TransactionLog</type>
      <defaultName>SellingTransactions-1.xml</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

Code Block 13 Sample configuration

2.5.9.3 Transaction Log Import - SAP CAR

Summary

This import is used to import transaction logs in XML format from the file(s) or from a CAR OData HTTP endpoint MultiChannelSalesQueryResults. Transaction logs are, for example, transactions from stationary POS systems or from an e-commerce system.

External Documentation

SAP Documentation:

https://help.sap.com/saphelp_car200/helpdata/en/5b/edc6533375eb1ae10000000a441470/frameapplicati onset.htm

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET		<ul style="list-style-type: none">Ad hoc reading of one or more files from the file system.Ad hoc querying of data from an endpoint via HTTP GET from CAR.
POST		<ul style="list-style-type: none">One single file or multiple files in a ZIP archive are transmitted via HTTP POST
PUT		<ul style="list-style-type: none">One single file or multiple files in a ZIP archive are transmitted via HTTP PUT
Scheduler		<ul style="list-style-type: none">Time-controlled reading of one or more files from the file system.Time-controlled querying of a file from an endpoint via HTTP GET from CAR.

Query

Example:

```
http://fiori-dev-hanadb01.gk.gk-
software.com:8000/sap/hba/t/rtl/car/int/odata/CARServices.xsodata/MultiChannelSalesQuery(P_SAPClient='5
42')/Results?$format=xml&$select=Article,BillingDocumentDate,Location,OrderChannel,NumberOfSalesItems,D
iscountAmount,NetSalesAmount,GenID&$top=500&$filter=Article ne '' and Article ne 'not_found' and
BillingDocumentDate gt '20181110'
```

Code Block 14 Sample input

Parameter	Sample Values	Remarks
\$format	xml	format of the return
\$select	Article,BilldingDocumentDate, ...	list of requested data separated by comma Required data for proper import: <ul style="list-style-type: none"> • Item • BillingDocumentDate • Location • OrderChannel • NumberOfSalesItems • DiscountAmount • NetSalesAmount • GenID
\$top	500	Maximum number of entries to be returned
\$filter	Item ne '' and item ne 'not_found'	filter for limiting the number of results Can only be applied to columns of the data type String. Possible arithmetic operators are: ne (not equal), eq (equal), lt (less than), gt (greater than). Possible logical operators are: and


Data

Example

```
<feed
  xml:base="http://fiori-dev-hanadb01.gk.gk-
software.com:8000/sap/hba/t/rtL/car/int/odata/CARServices.xsodata/">
  <title type="text">Results</title>
  <id>http://fiori-dev-hanadb01.gk.gk-
software.com:8000/sap/hba/t/rtL/car/int/odata/CARServices.xsodata/Results</id>
  <author>
    <name />
  </author>
  <link rel="self" title="Results" href="Results" />
  <entry>
    <id>http://fiori-dev-hanadb01.gk.gk-
software.com:8000/sap/hba/t/rtL/car/int/odata/CARServices.xsodata/MultiChannelSalesQueryResults('602486
32935337151')</id>
    <title type="text" />
    <author>
      <name />
    </author>
    <link rel="self" title="MultiChannelSalesQueryResults"
href="MultiChannelSalesQueryResults('60248632935337151')> />
    <category term="sap.hba.t.rtl.car.int.odata.CARServices.MultiChannelSalesQueryResultsType"
scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme" />
    <content type="application/xml">
      <m:properties>
        <d:GenID m:type="Edm.String">60248632935337151</d:GenID>
        <d:OrderChannel m:type="Edm.String" />
        <d:Location m:type="Edm.String" />
        <d:Article m:type="Edm.String">4711</d:Article>
        <d:BillingDocumentDate m:type="Edm.String">20181120</d:BillingDocumentDate>
        <d:NumberOfSalesItems m:type="Edm.Int32">1</d:NumberOfSalesItems>
        <d:NetSalesAmount m:type="Edm.Decimal">10</d:NetSalesAmount>
        <d:DiscountAmount m:type="Edm.Decimal">0</d:DiscountAmount>
      </m:properties>
    </content>
  </entry>
  <entry>
    ...
  </entry>
</feed>
```

Code Block 15 Sample input

Mapping and Validation

XML tag	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code
Item	String	4711	x	Item ID		<ul style="list-style-type: none"> not set: 51011
BillingDocumentDate	String yyyyMMdd	20181120	x	Timestamp of the transaction	CAR field includes only date and no time; Time is therefore set to 00:00:00 o'clock	<ul style="list-style-type: none"> not set: 51011 invalid value: 51015
Location	String			Store or shop	 not evaluated yet, because not implemented in POJO	
OrderChannel	String	web, walk-in...		The default channel can be defined in the configuration file of the Import Service.		
NumberOfSalesItems	Int32	1	x	NumberOfSalesItems is the quantity of the sold units minus the returns;	Entries where NumberOfSalesItems < 1 are not imported	<ul style="list-style-type: none"> not set: 51011 value < 1: 51018 invalid value: 51014
DiscountAmount, NumberOfSalesItems	Decimal	2.25		The discount is a result of the quotients from DiscountAmount over NumberOfSalesItems; The discount is rounded to two decimal places;		<ul style="list-style-type: none"> not set: 51011 invalid value or value < 0: 51016
NetSalesAmount, NumberOfSalesItems	Decimal	6.75	x	The price is a result of the quotient from the NetSalesAmount over NumberOfSalesItems; The price is rounded to two decimal places; NetSalesAmount is the net amount for all the transactions; It can be defined, in the configuration file of the Import Service, if the entries where NetSalesAmount = 0.00 should be imported (default: do not import)		<ul style="list-style-type: none"> not set: 51011 invalid value or value < 0: 51016
GenID	String	60248632935337151	x	automatically generated ID		<ul style="list-style-type: none"> not set: 51011

Configuration

General

Tag	Format	Default	Description
<sapClient>	String		The ID of the SAP tenant, is required for querying.
<defaultChannelId>	String		Channel ID that should be used when no channel is indicated in the transaction.
<includeZeroRevenues>	boolean		Should the transactions where Revenue = 0 be imported?
<excludeArticles> <Article>	String		Transactions for items with this name are not imported.
<importMode>	INITIAL, UPDATE	UPDATE	INITIAL = Deletes transaction data before the import starts. UPDATE = Updates existing entities and adds new entities.

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

Special Notes

Tag	Format	Description
<query>	String	Several parameters that are necessary for the query are supported in the query: <ul style="list-style-type: none">• \${SAP_CLIENT}: is filled with content from <sapClient>• \${FILTER_TOP}: is filled with content from <batchSize>, if > 0• \${FILTER_CONDITION}: is filled by the importer
<batchSize>	Integer	Amount of data that can be loaded at once. If the value is 0, no incremental loading occurs.

XML Reading

Please see details about the configuration [here](#).

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <importMode>UPDATE</importMode>
  <sapClient>400</sapClient>
  <minimumBusinessDay>20190201</minimumBusinessDay>
  <defaultChannelId>Store</defaultChannelId>
  <includeZeroRevenues>>false</includeZeroRevenues>
  <excludeArticles>
    <Article>not_found</Article>
    <Article>bad_Article_Id</Article>
  </excludeArticles>
  <xmlReader>
    <charset>UTF-8</charset>
  </xmlReader>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>TransactionLog</mappingId>
      <regExpression>SellingTransactions-.*</regExpression>
      <type>TransactionLog</type>
      <defaultName>SellingTransactions-default.xml</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <httpInboundAdapter>
    <waitOnDownloadFailedInSeconds>5</waitOnDownloadFailedInSeconds>
    <retryCounter>0</retryCounter>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <httpResources>
      <httpResource>
        <resourceId>MultiChannelSalesQuery</resourceId>
        <resourceUrl><![CDATA[http://sap-pq-hdb1.gk.gk-
software.com:8000/sap/hba/t/rtl/car/int/odata/CARServices.xsodata/MultiChannelSalesQuery(P_SAPC
lient='${SAP_CLIENT}')/Results?$format=xml&$select=BillingDocument,Article,BillingDocumentDate,
Location,OrderChannel,NumberOfSalesItems,DiscountAmount,NetSalesAmount,EuropeanArticleNumber,Sa
lesUnit${FILTER_TOP}&$orderby=BillingDocumentDate asc&$filter=Article ne '' and Article ne
'not_found' ${FILTER_CONDITION}]]></resourceUrl>
        <socketTimeoutInSeconds>30</socketTimeoutInSeconds>
        <connectionTimeoutInSeconds>200</connectionTimeoutInSeconds>
        <httpMethod>GET</httpMethod>
        <batchSize>0</batchSize>
        <type>TransactionLog</type>
        <authentication>
          <user>CARSERVICES</user>
          <encryptedPassword>TO_BE_FILLED</encryptedPassword>
        </authentication>
      </httpResource>
    </httpResources>
  </httpInboundAdapter>
</import-config>
```

Code Block 16 Sample configuration

2.5.9.4 Transaction Log RDE-Server

Summary

This import is used to import transaction logs from running RDE Servers via HTTP endpoint.

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
PUT		• One single file or multiple files in a ZIP archive are transmitted via HTTP PUT.
Scheduler		• Time-controlled querying of a file from an endpoint via HTTP PUT.

Data

A transaction represents an event. In the case of purchase orders, a transaction refers to a line item of the order. If several line items exist, these must be distributed with the same transaction ID in several transaction entries.

Mapping and Validation

CSV field	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code															
time	YYYY-MM-DD hh:mm:ss (ISO 8601 Semantics)	2018-01-05 15:45:33	x	Timestamp of the transaction in UTC time	Convert to timestamp (long value in seconds)	<ul style="list-style-type: none">not set: 51011															
transactID	String	928888281	x	Transaction ID. When transmitting several line items of a purchase order, an entry must be sent for each line item. The transaction ID may be the same for all line items.		<ul style="list-style-type: none">not set: 51011															
itemID	String	1245		Item ID		<ul style="list-style-type: none">not set: 51011															
transType	-2, -1, 0, 1	1	x	Transaction type. In addition to purchase orders, tracking transactions from the e-commerce system, too, such as search/click/shopping cart, can be used. Whether this information is used depends on the particular pricing strategy.	<table><tr><th>Value</th><th>Mapping</th><th>Meaning</th></tr><tr><td>-2</td><td>s</td><td>search</td></tr><tr><td>-1</td><td>c</td><td>Click</td></tr><tr><td>0</td><td>b</td><td>shopping cart</td></tr><tr><td>1</td><td>o</td><td>purchase order</td></tr></table>	Value	Mapping	Meaning	-2	s	search	-1	c	Click	0	b	shopping cart	1	o	purchase order	<ul style="list-style-type: none">not set: 51011invalid value: 51017
Value	Mapping	Meaning																			
-2	s	search																			
-1	c	Click																			
0	b	shopping cart																			
1	o	purchase order																			
basket	integer [1..max integer], if transType = 0, otherwise 0 or empty		(x)	For transaction type "shopping cart" only: Number of pieces of the item that was placed in the shopping cart.	Evaluated if transType = 0	<ul style="list-style-type: none">not set: 51011if transType is basket:value < 1: 51018value is not a valid number: 51014else:invalid value: 51017															
order	integer [1..max integer], if transType = 1, otherwise 0 or empty		(x)	For transaction type "purchase order" only: Number of pieces of the item that was placed in the shopping cart.	Evaluated if transType = 1	<ul style="list-style-type: none">not set: 51011if transType is order:value < 1: 51018value is not a valid number: 51014else:invalid value: 51017															
price	Floating point with "." as separator of decimal places. No thousands, maximum two decimal places.	15.99	(x)	For transaction types "purchase order"/"shopping cart" only: line item price (gross): Item price (not multiplied by quantity!). Assigned discounts are already deducted.	Evaluated if transType = 0 or = 1	<ul style="list-style-type: none">not set: 51011value is not a valid price or value < 0: 51016															
channelID	String	online		Name or ID of the sales channel. Is used, for example, to differentiate between offline/online channels.																	

Handling Timezones

The importer converts the time to the timezone of the business unit.

```
time|...  
2017-09-01 10:05:37|...  
Business Unit Timezone: GMT+02:00  
Is saved as 2017-09-01 12:05:37
```

Configuration

Tag	Format	Mandatory Field	Default	Description
minimumBusinessDay	yyyy-MM-dd		1970-01-01	Only transactions newer than this date will be imported.
importOrdersEnabled	true, false		true	Import order transactions?
importBasketsEnabled	true, false		false	Import basket transactions?
importSearchesEnabled	true, false		false	Import search transactions?
importClicksEnabled	true, false		false	Import click transactions?
importMode	INITIAL, UPDATE		UPDATE	INITIAL = Deletes transaction data before the import starts. UPDATE = Adds new entities.

Communication

This importer only supports HTTP endpoints and cannot be used to import from file system. It will remember what transactions have already been imported and in UPDATE mode only import new ones.

Import via HTTP Endpoint

Configuration

See "Import Communication Interface".

CSV Reading

Please see details about the configuration here.

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <importOrdersEnabled>true</importOrdersEnabled>
  <importSearchesEnabled>true</importSearchesEnabled>
  <importBasketsEnabled>true</importBasketsEnabled>
  <importClicksEnabled>true</importClicksEnabled>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>translogWithoutDate</mappingId>
      <regExpression>translog.*csv</regExpression>
      <type>TransactionLog</type>
      <defaultName>translog.csv</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>translogWithDate</mappingId>
      <regExpression>translog\..*</regExpression>
      <type>TransactionLog</type>
      <defaultName>translog.csv</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <csvReader>
    <charset>UTF-8</charset>
    <readHeader>true</readHeader>
    <escapeMode>BACKSLASH</escapeMode>
    <columnSeparator>|</columnSeparator>
  </csvReader>
  <httpInboundAdapter>
    <waitOnDownloadFailedInSeconds>5</waitOnDownloadFailedInSeconds>
    <retryCounter>0</retryCounter>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <httpResources>
      <httpResource>
        <resourceId>TransactionLog</resourceId>

<resourceUrl>http://localhost:8180/rde_server/admin/res/c101d7bda757/recommendations/export/transLog/archives
      </resourceUrl>
      <contentType>text/xml</contentType>
      <httpMethod>PUT</httpMethod>
      <batchSize>0</batchSize>
      <type>TransactionLog</type>
      <authentication>
        <user>admin</user>
        <encryptedPassword>admin</encryptedPassword>
      </authentication>
    </httpResource>
  </httpResources>
</httpInboundAdapter>
</import-config>
```

2.5.9.5 Transaction Log Import - SAP CSI

Summary

This import is used to import transaction logs from SAP CSI (Consumer Sales Intelligence).

You can find the original API for the SAP CSI service here:
<https://api.sap.com/api/TLOGCollectorAPI/resource>

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET		<ul style="list-style-type: none"> Ad hoc reading of one or more files from the file system. Ad hoc querying of data from an endpoint via HTTP GET.
POST		<ul style="list-style-type: none"> One single file or multiple files in a ZIP archive are transmitted via HTTP POST.
PUT		<ul style="list-style-type: none"> One single file or multiple files in a ZIP archive are transmitted via HTTP PUT.
Scheduler		<ul style="list-style-type: none"> Time-controlled reading of one or more files from the file system. Time-controlled querying of a file from an endpoint via HTTP GET, PUT or POST.

Data


```
[
  {
    "transactionId": "",
    "retailStoreId": "R110",
    "businessDayDate": "2018-09-24",
    "workstationId": "0000000010",
    "sequenceNumber": 445566,
    "beginDateTime": "2018-09-24T08:25:41Z",
    "endDateTime": "2018-09-24T08:28:32Z",
    "transactionIndex": 29,
    "department": "000000002",
    "operatorId": "Operator1",
    "currencyCode": "USD",
    "trainingFlag": false,
    "keyedOfflineFlag": false,
    "cancelFlag": false,
    "voidFlag": false,
    "postVoidFlag": false,
    "creationDateTime": "2018-09-24T13:26:26.30563Z",
    "creationUser": "User1",
    "lastUpdateDateTime": "",
    "lastUpdateUser": "",
    "blockingFlag": false,
    "packageId": "",
    "retailTransactionTypeCode": "1001",
    "channel": "04",
    "retailLineItems": [
      {
        "sequenceNumber": 1,
        "itemIdTypeCode": "",
        "itemId": "MaterialNumber1",
        "quantity": "1.000",
        "quantityUnitOfMeasure": "ST",
        "salesAmount": "68.64",
        "normalSalesAmount": "68.64",
        "actualUnitPrice": "68.64",
        "units": "1.000",
        "serialNumber": "2000067890987876890",
        "voidFlag": false,
        "voidedLine": "0",
        "discounts": [
          {
            "sequenceNumber": "",
            "discountTypeCode": "",
            "amount": "",
            "discountId": "",
            "previousPrice": "",
            "newPrice": ""
          }
        ],
        "taxes": [
          {
            "sequenceNumber": "",
            "taxTypeCode": "",
            "amount": "",
            "taxableAmount": "",
            "taxIncludedInTaxableAmountFlag": "",
            "percent": "",
            "taxGroupId": "",
            "postTaxGroupId": ""
          }
        ],
        "warrantyCode": ""
      }
    ],
    "discounts": [
      {
        "sequenceNumber": "",
        "discountTypeCode": "",
        "amount": "",
        "discountId": "",
        "previousPrice": "",
        "newPrice": ""
      }
    ],
    "taxes": [
      {

```

```

        "sequenceNumber": 1,
        "taxTypeCode": "4301",
        "amount": 18.64,
        "taxableAmount": 50,
        "taxIncludedInTaxableAmountFlag": true,
        "percent": 0,
        "taxGroupId": "",
        "postTaxGroupId": ""
    }
],
"tenders": [
    {
        "sequenceNumber": 1,
        "tenderTypeCode": "3101",
        "tenderId": "YY01",
        "amount": 68.64,
        "foreignCurrency": "NA",
        "tenderCategory": "A",
        "tenderSubCategory": "A",
        "foreignAmount": 0,
        "foreignAmountExchangeRate": 0,
        "cancelFlag": false,
        "typeCode": "",
        "linkTenderSequenceNumber": ""
    }
],
"customer": {
    "customerId": "",
    "customerType": ""
},
"postVoidHeader": {
    "postVoidTypeCode": "",
    "voidRetailStoreId": "",
    "voidBusinessDayDate": "",
    "voidWorkstationId": "",
    "voidSequenceNumber": "",
    "voidBeginDateTime": ""
},
"referenceFileName": "",
"receiptNumber": ""
}
]

```

Code Block 17 Sample input

Mapping and Validation

Each line item of a transaction is mapped to a dynamic pricing transaction. Only sales transactions/line items are mapped.

When is a Line Item a Sales Transaction?

All conditions have to be true:

- On transaction level
 - Not cancelled: cancelFlag = false
 - No training mode: trainingFlag = false
 - No void: voidFlag = false and postVoidFlag = false
 - Valid retailTransactionTypeCode: in configured list
- On retail line item level
 - No void: voidFlag = false, voidedLine = 0
 - Valid lineItemTypeCode: in configured list

Mapping of a Line Item

JSON Field	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code
transactionId		1212	x	ID of the retail line item transaction	transactionId + retailLineItems.sequenceNumber	• not set: 51011
beginTransactionTime	YYYY-MM-DD hh:mm:ss (ISO 8601 Semantics)	2018-01-05T15:45:33Z	x	Timestamp of the transaction	time zone of the business unit and configuration need to be taken into account for conversion. See handling timezones in the next section.	• not set: 51011
retailLineItems.itemID	String	1245		Item ID	itemID	• not set: 51011
	1			Order	transType	
	0				basket	
retailLineItems.quantity	number minimum: -999999999.999 maximum: 999999999.999 The quantity of the item sold in the transaction. Quantity is conveyed in Decimal(13,3) format where 13 specifies the precision or the number of total digits (the sum of whole digits and fractional digits), and 3 specifies the number of fractional digits.	1.000	x	If retailLineItems.quantityUnitOfMeasure = configured value of quantityUnitOfMeasurePiece: Then convert to integer Otherwise: use 1	order	
retailLineItems.salesAmount	number minimum: -999999999.99 maximum: 999999999.99	15.99	x	The amount charged for an item sold in a transaction. Sales Amount is conveyed in Decimal(15,2) format where 15 specifies the precision or the number of total digits (the sum of whole digits and fractional digits), and 2 specifies the number of fractional digits.	price	• not set: 51011 • Not a valid price or price < 0: 51016
		14.99		Line item price (net, without taxes): Item price (not multiplied by quantity!). Assigned discounts are already deducted.	retailLineItems.salesAmount - (retailLineItems.actualUnitPrice * retailLineItems.units) netPrice = promotionAmount - (SUM for all taxes.amount)	
		1.23		Sum of line item discounts or total discounts granted for this line item, for one item (i.e. without considering the quantity).	promotionAmount = retailLineItems.salesAmount - (retailLineItems.actualUnitPrice * retailLineItems.units)	

channel	String	04		Name or ID of the sales channel	channelID If channel is empty, use the configured value from the configuration. If this is not set, do not set a channel	
---------	--------	----	--	---------------------------------	---	--

Handling Timezones

The importer uses the timezone that is passed in the beginDateTime field to interpret the time, converts it to the business unit timezone and saves it.

Example:

beginDateTime: 2018-07-31T17:25:34.000+02:00

Business Unit Timezone: GMT+02:00

Is saved as 2018-07-31 17:25:34.000.

Configuration

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

General Settings

Tag	Format	Mandatory Field	Default	Example	Description
importMode	INITIAL, UPDATE		UPDATE		INITIAL = Deletes transaction data before the import starts. UPDATE = Updates existing entities and adds new entities.
channelID	String				Used for the channel if the transaction data does not provide a channel or it is empty.
salesRetailTransactionTypeCodes	Comma separated Strings	x		1001,1004,1005	Only transactions with this type code are imported, others are skipped
salesLineItemtypeCodes	Comma separated Strings	x		2001	Only line items with this type code are imported, others are skipped
quantityUnitOfMeasurePiece	Comma separated Strings	x		EA,PCE	UOMs used for a piece (e.g. EA for each, ST for Stück). Other UOMs get quantity set to 1
supportedItemIdTypeCodes	Comma separated Strings	x		1	1 = GTIN is used as item id in pricing as well, so we can do a lookup without conversion. So we use the GTIN also as item id in pricing. Other variants (material id, etc.) are currently not supported.

Import via File System

Configuration

See "Import Communication Interface".

Import via HTTP Endpoint

Configuration

See "Import Communication Interface".

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <channelID>Store</channelID>
  <validRetailTransactionTypeCodes>1001</validRetailTransactionTypeCodes>
  <validLineItemTypeCodes>10,12</validLineItemTypeCodes>
  <quantityUnitOfMeasurePiece>ST</quantityUnitOfMeasurePiece>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>TransactionLog</mappingId>
      <regExpression>SellingTransactions-.*</regExpression>
      <type>TransactionLog</type>
      <defaultName>SellingTransactions-1.xml</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

Code Block 18 Sample configuration

2.5.10 Master Data Import

2.5.10.1 Item Import - AddOn CSV

Summary

This import is used to evaluate additional item master data in CSV format from a file or from an HTTP endpoint.

This import may be used to add more data per item, when that data was not evaluated during the main import of master data, e.g.:

- Price lower/upper limit
- Parameter

Update Strategy

For each record the importer looks for an existing record that matches this ID. If there is no existing record for this ID and error is reported for that item. If the item exists, all fields are updated that are part of the record.

If you add a column in the header of the CSV but don't fill it for the record it depends on the field whether it is set to a default (e.g. booleans are set to false) or the value is removed at all (e.g. "brand"). Please check the column "Default if empty" what's done for each field.

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET	Item update	Ad hoc reading of the file from the file system or from an HTTP endpoint
POST	Item update	The file is transferred as ZIP archive
Scheduler	Item update	Time-controlled reading of the file from the file system or by an HTTP endpoint

Data

```
pid|minUnitPrice|maxUnitPrice
000019595|20.00|30.00
000019613|30.00|50.00
```


CSV field	Format	Example	Mandatory Field	Description	Default if empty	Validation and Error Code
pid	String	112344	x	Item ID	Error	<ul style="list-style-type: none"> not set: 51011
regularUnitPrice	Number in US format, no thousands separator, "." for comma	12.99		Normal price of the item	Removed	<ul style="list-style-type: none"> invalid format or value < 0: 51016
minUnitPrice	Number in US format, no thousands separator, "." for comma	10.1		Minimum price of the item. The pricing strategies use minPricingLimit. This minimum price is the absolute lower limit that is also not allowed to be exceeded by minPricingLimit.	Removed	<ul style="list-style-type: none"> not set: 51011 invalid format or value <= 0: 51016 invalid pricing limit: 51024
maxUnitPrice	Number in US format, no thousands separator, "." for comma	11		Maximum price of the item. The pricing strategies use maxPricingLimit. This maximum price is the absolute upper limit that is also not allowed to be exceeded by maxPricingLimit.	Removed	<ul style="list-style-type: none"> not set: 51011 invalid format or value <= 0: 51016 invalid pricing limit: 51024
expiryDate	Date: yyyy-MM-dd	2018-12-01		Date for sales	Removed	<ul style="list-style-type: none"> invalid format: 51015
onlineFlag	0, 1	1		Is the item sellable or not? 0 = false 1 = true (Is set to 1 by standard importers.)	0 = false	<ul style="list-style-type: none"> invalid format: 51017
stock	Long	10		Current stock	0	<ul style="list-style-type: none"> value is not a valid number: 51022
name	String			Item name	Removed	
brand	String			Item brand	Removed	
promotionUnitPrice	Number in US format, no thousands separator, "." for comma			Promotion price	Removed	<ul style="list-style-type: none"> invalid format or value < 0: 51016
masterID	String	12345			Removed	<ul style="list-style-type: none"> item does not exist: 51020
minPricingLimit	Number in US format, no thousands separator, "." for comma	99.00		Lower limit for the price algorithm.	Removed	<ul style="list-style-type: none"> not set: 51011 invalid format or value <= 0: 51016 invalid pricing limit: 51024
maxPricingLimit	Number in US format, no thousands separator, "." for comma	129.50		Upper limit for the price algorithm.	Removed	<ul style="list-style-type: none"> not set: 51011 invalid format or value <= 0: 51016 invalid pricing limit: 51024

purchaseUnitPrice	Number in US format, no thousands separator, "." for comma	50.12		Purchase price	Removed	<ul style="list-style-type: none"> invalid format or value <= 0: 51016
recommendedRetailPrice	Number in US format, no thousands separator, "." for comma	55.99		Non-committal price recommendation	Removed	<ul style="list-style-type: none"> invalid format or value < 0: 51016
measureUnitCode	String	PCE		Unit of measure	Removed	
materialId	String	98387377		SAP Material ID. In SAP data model, an item is addressed by the combination materialid and unit of measure. For SAP price export we need the material id to set the price on SAP side.	Removed	
quantity	Number in US format, no thousands separator, "." for comma	12		Quantity	Removed	<ul style="list-style-type: none"> invalid format: 51016
pricingenabled	0, 1	0		Use optimized pricing for this product? 0 = false 1 = true	0 = false	<ul style="list-style-type: none"> invalid format: 51012
newproduct	0, 1	1		Is this a new product (for initial pricing)? 0 = false 1 = true	0 = false	<ul style="list-style-type: none"> invalid format: 51012
basequantityunit	String			Base Quantity Unit		
imageUrl	String	http://xxx.jpg		URL to product image		
itemTypeCode	1 = Standard 2 = Master 3 = Variant 4 = Set	1		Type of the item.		<ul style="list-style-type: none"> Invalid value: 51017

You can add your own data fields via the SDK extensions. This is the preferred way to add new data as you can then see data field names in filters and in the master data explorer.

If you don't need this, please install the "Dynamic Pricing Extension" plugin that supports data fields param1 - param20, also for importing data via this import and the item masterdata CSV import.

For all optional fields you can force them to be treated as mandatory fields by configuration. See here for details about configuration.

Example

```
pid|netUnitPrice|name|strikeOutPrice|minUnitPrice|maxUnitPrice
000019595|26.90|THE SOFT KHAKE SLIM FIT|22.46|20.00|30.00
000019613|39.90|THE SOFT KHAKE SLIM FIT|30.00|50.00
```

Configuration

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

General

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

CSV Reading

Please see details about the configuration here.

Data Mapping

This import supports mapping of CSV data fields to internal data fields and fixed values via XML configuration.

This import supports mapping of fields of those entities:

Entity	<type>
Item	Item

For more details about the mapping, please refer to the documentation here.

Mapping Example

```
<fieldMappings>
  <fieldMapping>
    <type>Item</type>
    <mode>ColumnName</mode>
    <sourceField>netUnitPrice</sourceField>
    <destinationField>regularUnitPrice</destinationField>
  </fieldMapping>
</fieldMappings>
```

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>ItemAddon</mappingId>
      <regExpression>itemaddon.*csv</regExpression>
      <type>ItemAddon</type>
      <defaultName>itemaddon.csv</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <csvReader>
    <charset>UTF-8</charset>
    <readHeader>true</readHeader>
    <escapeMode>BACKSLASH</escapeMode>
    <columnSeparator>|</columnSeparator>
  </csvReader>
  <!--fieldMappings>
    <fieldMapping>
      <type>Item</type>
      <mode>ColumnName</mode>
      <sourceField>netUnitPrice</sourceField>
      <destinationField>regularUnitPrice</destinationField>
    </fieldMapping>
  </fieldMappings-->
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

itemTypeCode

2.5.10.2 Master Data Import - CSV

Summary

This import is used to import master data in CSV format from files. For example:

- Items and prices
- Categories
- Category hierarchy/hierarchies
- Item assignments

The service supports both an initial mode that deletes all data before starting the import and an update mode.

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET	Previous deletion of the current data	Ad hoc reading of the files from the file system or by an HTTP endpoint
POST	Previous deletion of the current data	The files in the ZIP archive are transferred
Scheduler	Previous deletion of the current data	Time-controlled reading of the files from the file system or from an HTTP endpoint

Mapping and Validation

Taxonomies

Several taxonomies (separated category trees) are supported for item category assignments and category hierarchies.

If no taxonomy is indicated, the default category indicated in the configuration will be used. If no taxonomy has been created yet, it will be generated automatically.

A taxonomy is created by prepending it to the file name of the category hierarchy and category assignment file. If this taxonomy already exists, the data will be assigned to it.

Sample taxonomy "zubehoer":

- zubehoer.itemcategories.csv
- zubehoer.categoryhiearchy.csv

Categories

CSV field	Format	Example	Mandatory Field	Description	Database Field	Mapping	Validation and Error Code
cid	String	10002	x	ID of the category	categories.cid		<ul style="list-style-type: none">• not set: 51011• cid does already exist: 51029
cname	String	Shoes		Name of the category	categories.name		

Example

```
cid|cname
10000|
10001|Schuhe
```

Category Hierarchy

CSV field	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code
cid	String	10002	x	ID of the category		<ul style="list-style-type: none">not set: 51011category does not exist: 51019
cidParent	String	10001	x	ID of the parent category		<ul style="list-style-type: none">not set: 51011category does not exist: 51019

Example

```
cid|cidParent
10001|10000
10002|10000
10003|10002
```

Item Category Assignment

CSV field	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code
pid	String	12345	x	Item ID		<ul style="list-style-type: none">not set: 51011item does not exist: 51020
cid	String	10001	x	Category ID of the item		<ul style="list-style-type: none">not set: 51011category does not exist: 51019

Example

```
pid|cid
123456|10000
298383|10000
111322|10002
```

Item

CSV field	Format	Example	Mandatory Field	Default	Description	Mapping	Validation and Error Code	Slow/Fast Mover
pid	String	112344	x		Item ID		<ul style="list-style-type: none"> not set: 51011 	
regularUnitPrice	Number in US format, no thousands separator, "." for comma	12.99			Normal price of the item		<ul style="list-style-type: none"> invalid format or value < 0: 51016 	Fast
minUnitPrice	Number in US format, no thousands separator, "." for comma	10.1			Minimum price of the item. The pricing strategies use minPricingLimit. This minimum price is the absolute lower limit that is also not allowed to be exceeded by minPricingLimit.		<ul style="list-style-type: none"> not set: 51011 invalid format or value <= 0: 51016 invalid pricing limit: 51024 	Slow
maxUnitPrice	Number in US format, no thousands separator, "." for comma	11			Maximum price of the item. The pricing strategies use maxPricingLimit. This maximum price is the absolute upper limit that is also not allowed to be exceeded by maxPricingLimit.		<ul style="list-style-type: none"> not set: 51011 invalid format or value <= 0: 51016 invalid pricing limit: 51024 	Slow
expiryDate	Date: yyyy-MM-dd	2018-12-01			Date for sales		<ul style="list-style-type: none"> invalid format: 51015 	Slow
onlineFlag	0, 1	1			Is the item sellable or not? 0 = false 1 = true	If not set, the value 1 is assumed. 0 = offline, 1 = online	<ul style="list-style-type: none"> invalid format: 51017 	Fast
stock	Long	10			Current stock	If not set, stock = 0	<ul style="list-style-type: none"> value is not a valid number: 51022 	Fast
name	String				Item name			Slow
brand	String				Item brand			Slow
promotionUnitPrice	Number in US format, no thousands separator, "." for comma				Promotion price		<ul style="list-style-type: none"> invalid format or value < 0: 51016 	Slow

masterID	String	12345					<ul style="list-style-type: none"> If masterID mandatory is enforced and referenced item is not part of the import: item does not exist: 51020 	Slow
minPricingLimit	Number in US format, no thousands separator, "." for comma	99.00			Lower limit for the price algorithm.	<ul style="list-style-type: none"> If not set yet, it will be set to minUnitPrice. 	<ul style="list-style-type: none"> not set: 51011 invalid format or value <= 0: 51016 invalid pricing limit: 51024 	Slow
maxPricingLimit	Number in US format, no thousands separator, "." for comma	129.50			Upper limit for the price algorithm.	<ul style="list-style-type: none"> If not set yet, it will be set to maxUnitPrice. 	<ul style="list-style-type: none"> not set: 51011 invalid format or value <= 0: 51016 invalid pricing limit: 51024 	Slow
purchaseUnitPrice	Number in US format, no thousands separator, "." for comma	50.12			Purchase price		<ul style="list-style-type: none"> invalid format or value <= 0: 51016 	Slow
recommendedRetailPrice	Number in US format, no thousands separator, "." for comma	55.99			Non-committal price recommendation		<ul style="list-style-type: none"> invalid format or value < 0: 51016 	Slow
measureUnitCode	String	PCE			Unit of measure If baseQuantityUnit is set, but this field is empty, then use baseQuantityUnit			Slow

materialId	String	98387377			SAP Material ID. In SAP data model, an item is addressed by the combination materialId and unit of measure. For SAP price export we need the material id to set the price on SAP side.			Slow
quantity	Number in US format, no thousands separator, "." for comma	12		1	Quantity		<ul style="list-style-type: none"> invalid format: 51016 	Slow
pricingenabled	0, 1	0			Use optimized pricing for this product? 0 = false 1 = true		<ul style="list-style-type: none"> invalid format: 51012 	Slow
newproduct	0, 1	1			Is this a new product (for initial pricing)? 0 = false 1 = true		<ul style="list-style-type: none"> invalid format: 51012 	Slow
baseQuantityUnit	String				Base Quantity Unit If measureUnitCode set and this field is empty, then use measureUnitCode.			Slow
imageUrl	String	http://xxxx.jpg			URL to a product image			Slow
itemTypeCode	1 = Standard 2 = Master 3 = Variant 4 = Set	1	(x)		Type of the item.	The behavior of the import depends on the configuration of <itemTypeImportMode>. It can be optional or mandatory.	<ul style="list-style-type: none"> Invalid value: 51017 	Slow

Differences to RDE items.csv format:

RDE format	Pricing format
netUnitPrice	regularUnitPrice
netPurchasePrice	purchaseUnitPrice
masterUID	masterID
strikeOutPrice	promotionUnitPrice

Custom Fields by Extension

You can add your own data fields via the SDK extensions. This is the preferred way to add new data as you can then see data field names in filters and in the master data explorer.

If you don't need this, please install the "Dynamic Pricing Extension" plugin that supports data fields param1 - param20, also for importing data via this import and the item addon CSV import.

Mandatory/Optional Behavior

For all optional fields you can force them to be treated as mandatory fields by configuration. See here for details about configuration.

Handling Master/Variants

If a variant references a master item by setting the ID in the masterID field, this master item does not have to be part of the import.

You can enforce that referenced master items have to be part of the import by configuration:

```
<fieldMapping>
  <type>Item</type>
  <mode>EnforceMandatory</mode>
  <destinationField>masterID</destinationField>
</fieldMapping>

or:

<fieldMapping>
  <type>Item</type>
  <mode>ColumnName</mode>
  <sourceField>myMaster</sourceField>
  <destinationField>masterID</destinationField>
  <mandatory/>
</fieldMapping>
```

Old behavior in releases before 3.1.0:

Before 3.1.0 it was mandatory to add the master item to the import stream if it was referenced by a variant. This behavior can be worked around only via custom extension SDK.

Example

```
pid|regularUnitPrice|name|strikeOutPrice|minUnitPrice|maxUnitPrice
000019595|26.90|THE SOFT KHAKI SLIM FIT|22.46|20.00|30.00
000019613|39.90|THE SOFT KHAKI SLIM FIT||30.00|50.00
```

Configuration

General

```
<defaultTaxonomy>default</defaultTaxonomy>
<importMode>INITIAL</importMode>
```

Tag	Format	Mandatory	Default	Example	Description
defaultTaxonomy	String	x			This taxonomy is used (and, where applicable, created) when no taxonomy is prepended to file name of the category assignment/category hierarchy file.
importMode	INITIAL UPDATE		INITIAL		<ul style="list-style-type: none"> INITIAL: Deletes masterdata before the import starts: Items, item categories, categories, category hierarchies. UPDATE: Updates existing entities and adds new entities. Updating does not mean, that the fields of an existing record are updated but that the record for this ID is overridden by all the new values coming in. If you send less columns for the update, only those columns are written, other columns are not left untouched but deleted.
itemTypeImportMode	OPTIONAL MANDATORY STANDARD INHERIT		OPTIONAL	MANDATORY	<ul style="list-style-type: none"> OPTIONAL: the item type code column is optional, so old import files are compatible. However, the itemTypeCode field is not filled MANDATORY: the itemTypeCode column is mandatory. So you need to set it via field mapping (fixedValue) or via CSV column data STANDARD: the itemTypeCode column in the data is not used but a fixed value of 1 (= Standard Item) for all items INHERIT: The itemTypeCode values are inherited from the values in the column masterID: <ul style="list-style-type: none"> If masterID is set, then this item is a variant: itemTypeCode = 3 If an item is referenced by a variant having its item id as masterID, then the is a master with itemTypeCode = 2 For all non referenced items having empty masterID column set the item to a standard item with itemTypeCode = 1

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

General

See "Import Communication Interface".

Sorting

The file names within a group are sorted alphabetically in ascending order.

File Name for Taxonomies

Sample taxonomy "zubehoer":

- zubehoer.itemcategories.csv
- zubehoer.categoryhiearchy.csv

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

CSV Reading

Please see details about the configuration here.

Data Mapping

This import supports mapping of CSV data fields to internal data fields and fixed values via XML configuration.

This import supports mapping of fields of those entities:

Entity	<type>
Item	Item
Category	Category
Item Category	ItemCategory
Category Hierarchy	CategoryHierarchy

For more details about the mapping, please refer to the [documentation here](#).

Mapping Example

```

<fieldMappings>
  <fieldMapping>
    <type>Item</type>
    <mode>ColumnName</mode>
    <sourceField>netUnitPrice</sourceField>
    <destinationField>regularUnitPrice</destinationField>
  </fieldMapping>
  <fieldMapping>
    <type>Category</type>
    <mode>ColumnName</mode>
    <sourceField>category</sourceField>
    <destinationField>cid</destinationField>
  </fieldMapping>
  <fieldMapping>
    <type>CategoryHierarchy</type>
    <mode>ColumnName</mode>
    <sourceField>parent</sourceField>
    <destinationField>cidParent</destinationField>
  </fieldMapping>
  <fieldMapping>
    <type>ItemCategory</type>
    <mode>ColumnName</mode>
    <sourceField>categoryId</sourceField>
    <destinationField>cid</destinationField>
  </fieldMapping>
</fieldMappings>

```

Example

```

<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <importMode>INITIAL</importMode>
  <defaultTaxonomy>default</defaultTaxonomy>
  <csvReader>
    <charset>UTF-8</charset>
    <readHeader>true</readHeader>
    <escapeMode>BACKSLASH</escapeMode>
    <columnSeparator>|</columnSeparator>
  </csvReader>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>Item</mappingId>
      <regExpression>items.*csv</regExpression>
      <type>Item</type>
      <defaultName>items.csv</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>Category</mappingId>
      <regExpression>categories.*csv</regExpression>
      <type>Category</type>
      <defaultName>category.csv</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>ItemCategory</mappingId>
      <regExpression>itemcategories.*csv</regExpression>
      <type>ItemCategory</type>
      <defaultName>itemcategories.csv</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>CategoryHierarchy</mappingId>
      <regExpression>categoryhierarchy.*csv</regExpression>
      <type>CategoryHierarchy</type>
      <defaultName>categoryhierarchy.csv</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>ItemCategoryWithTaxonomy</mappingId>
      <regExpression>.*\..itemcategories.*csv</regExpression>
      <type>ItemCategory</type>
      <defaultName>itemcategories.csv</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>CategoryHierarchyWithTaxonomy</mappingId>
      <regExpression>.*\..categoryhierarchy.*csv</regExpression>
      <type>CategoryHierarchy</type>
      <defaultName>categoryhierarchy.csv</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <fieldMappings>
    <fieldMapping>
      <type>Item</type>
      <mode>ColumnName</mode>
      <sourceField>netUnitPrice</sourceField>
      <destinationField>regularUnitPrice</destinationField>
    </fieldMapping>
    <fieldMapping>
      <type>Category</type>
      <mode>ColumnName</mode>
      <sourceField>category</sourceField>
      <destinationField>cid</destinationField>
    </fieldMapping>
    <fieldMapping>
      <type>CategoryHierarchy</type>
      <mode>ColumnName</mode>
      <sourceField>parent</sourceField>
      <destinationField>cidParent</destinationField>
    </fieldMapping>
    <fieldMapping>
      <type>ItemCategory</type>
      <mode>ColumnName</mode>
      <sourceField>categoryId</sourceField>
      <destinationField>cid</destinationField>
    </fieldMapping>
  </fieldMappings>
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
  </fileInboundAdapter>
</import-config>

```

```

<copyOption>NONE</copyOption>
<isLockFileEnabled>false</isLockFileEnabled>
<lockFileName>lock</lockFileName>
<triggerOption>NO_TRIGGER</triggerOption>
  <triggerFileName>trigger</triggerFileName>
<waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
<fileValidationRetryCount>0</fileValidationRetryCount>
</fileInboundAdapter>
</import-config>

```

Code Block 19 Sample configuration

2.5.10.3 Master Data Import - SAP MerchandiseERPReplicationBulkRequest

Summary

This import is used to import master data in XML format from the SAP ERP system via the Enterprise Services from a file or HTTP endpoint. SAP ERP/S4 uses the POST trigger to pass the data.

Service MerchandiseERPReplicationBulkRequest

- Items and prices
- Item assignments

Service ProductCategoryHierarchyERPReplicationRequest:

- Categories
- Category hierarchy(ies)
- Taxonomies

The service supports both an initial mode that deletes all data before starting the import and an update mode.

Import Details

Item Identification

SAP provides two IDs:

- Material ID
- Product Standard ID

These IDs are mapped to:

- Material ID → Material ID
- Product Standard ID → pid

Master/Variants

The tag MaterialMerchandiseTypeCode is processed to create master/variant relationships:

Value	Type	Action
1	Single Item	Imported as single item
2	Variant	Imported as variant, masterUID is filled with the master's Product Standard ID (not Material ID!)
3	Master	Imported as variant with empty masterUID
4	Set	Imported as set item

Taxonomy and Category Assignment

The importer supports assigning more than one category to an item. All product categories are processed that provide a ProductCategoryHierarchyID.

```
<ProductCategory>
  <InternalID>RF12111</InternalID>
  <ProductCategoryHierarchyID>1</ProductCategoryHierarchyID>
  <ProductCategoryHierarchyTypeCode>3</ProductCategoryHierarchyTypeCode>
</ProductCategory>
```

This ID is treated as a taxonomy and has to match the taxonomy that is imported with the category importer, here it is the tag <ID>:

```
<ProductCategoryHierarchy Actioncode="04">
  <ProductCategoryHierarchyTypeCode>3</ProductCategoryHierarchyTypeCode>
  <CompleteTransmissionIndicator>false</CompleteTransmissionIndicator>
  <ID>1</ID>
...

```

ProductCategoryHierarchyTypeCode is not processed in any way.

Handling Future Prices

There is a configuration switch to enable the import of future prices: **futurePriceImportEnabled**.

Future Prices import disabled

In case the import is disabled, no future prices are imported. However, also prices that do not have a valid from/to (StartDate, EndDate) date information in the price list, won't be imported.

To import a price the date of the import must be in a valid from/to interval of the price definition.

Examples:

- Settings:
 - Import Date = 08.07.2021
 - futurePriceImportEnabled = false
- Example 1:
 - StartDate = 03.01.2020
 - EndDate = 05.01.2020
 - **No price is imported**
- Example 2:
 - StartDate = 03.01.2020
 - EndDate = 05.01.2022
 - **Price is imported**

Future Prices import enabled

If future prices are enabled you can fine tune, if a price is imported as a future price or not and if the valid to boundary is open or not.

Valid to Boundary

Future prices can be stored as:

- Valid to a concrete date
- Valid to forever

To distinguish between these two cases, the importer needs to know what "forever" means. You can configure the concrete date for it in the configuration parameter **infinityDate**.

Examples:

- Settings:
 - infinityDate = 31.12.2099

- futurePriceImportEnabled = true
- Example 1:
 - StartDate = 03.01.2020
 - EndDate = 31.12.2025
 - **Valid to a concrete date: 30.12.2025**
- Example 2:
 - StartDate = 03.01.2020
 - EndDate = 31.12.2099
 - **Validity is unbounded**

When does the future begin?

You might want to import future prices, but maybe not all prices are future prices that just start tomorrow.

For instance you import the master data on 08.07. and the EffectiveDate is 09.07.21. In this case the importer creates a future price record. If you don't want that, you can tell the importer with the parameter **nowOffset** to move the start of the future to the next day.

- Settings:
 - infinityDate = 31.12.2099
 - futurePriceImportEnabled = true
 - nowOffset = 1
- Example 1:
 - Date of import = 08.07.2021
 - StartDate = 09.07.2021
 - EndDate = 31.12.2099
 - 1 day is added to the day of import, so 09.07.2021 is treated as now. For any prices that are getting effective now to infinity (see infinityDate parameter to control that), **no future price is created but just the price is set as regular price.**
- Example 2:
- Date of import = 08.07.2021
- StartDate = 09.07.2021
- EndDate = 31.12.2022
- 1 day is added to the day of import, so 09.07.2021 is treated as now. However, as the expiry date is not infinity, **a future price record is created with a valid to 31.12.2022.**

Unit of Measure

The following information is imported about unit of measures:

Data	Description
Unit of measure	Unit of measure of the item
Base unit of measure	Base unit. In this unit the stock is managed
Quantity in base unit of measure	Quantity in base unit of measure

Example for a box of coca cola containing 12 bottles:

Data	Example
Unit of measure	BOX
Base unit of measure	BOTTLE
Quantity in base unit of measure	12

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET	Previous deletion of the current data	Ad hoc reading of the files from the file system or by an HTTP endpoint
POST	Previous deletion of the current data	The files in the ZIP archive are transferred
Scheduler	Previous deletion of the current data	Time-controlled reading of the files from the file system or from an HTTP endpoint

Mapping and Validation

Item and Category

All the unnecessary spaces are removed before processing the values.

ERP field	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code
<MerchandiseERPReplicationRequestMessage> <Merchandise> <GlobalTradeItemNumber> <ProductStandardID>	String	2050000000102	x	Item Id		<ul style="list-style-type: none"> not set: 51011 item does already exist: 51029
<MerchandiseERPReplicationRequestMessage> <Merchandise> <Description> <Description languageCode="xyz">	String	writing case		Item Name	Only the items with the LanguageCode indicated in the Configuration file.	
<MerchandiseERPReplicationRequestMessage> <Merchandise> <SalesPriceInformation> <PriceSpecification> <Amount>	String x.yz	5.99	x	Sales price	If there is a validity period for a price (see ValidityPeriod) that fits the date of the import, the first price is imported. If there are future prices, they are imported as future price records. For details see "Handling of Future Prices".	<ul style="list-style-type: none"> not set: 51011 invalid value or value < 0: 51016
<MerchandiseERPReplicationRequestMessage> <Merchandise> <ProcurementPriceInformation> <PriceSpecification> <Amount>	String x.yz	2.84		Purchase price	Only the price that is valid according to the following time range is imported: <MerchandiseERPReplicationRequestMessage> <Merchandise> <ProcurementPriceInformation> <PriceSpecification> <ValidityPeriod> If there is a PriceSpecification without a ValidityPeriod, it will always have precedence. > 0	<ul style="list-style-type: none"> not set: 51011 invalid value or value < 0: 51016
<MerchandiseERPReplicationRequestMessage> <Merchandise> <SalesPriceInformation> <PriceSpecification> <ValidityPeriod> <StartDate>	String yyyy-MM-dd	1999-10-20		Valid from date of validity of the sales price	See "Handling Future Prices"	<ul style="list-style-type: none"> invalid value: 51015
<MerchandiseERPReplicationRequestMessage> <Merchandise> <SalesPriceInformation> <PriceSpecification> <ValidityPeriod> <EndDate>	String yyyy-MM-dd	1999-10-20		Expiration date of validity of the sales price	See "Handling Future Prices"	<ul style="list-style-type: none"> invalid value: 51015
<MerchandiseERPReplicationRequestMessage> <Merchandise> <ProductCategory> <InternalID>	String	1025		Category ID		<ul style="list-style-type: none"> not set: 51011 category does not exist: 51019
<MerchandiseERPReplicationRequestMessage> <Merchandise> <ProductCategory> <ProductCategoryHierarchyID>	String	1		Taxonomy		<ul style="list-style-type: none"> not set: 51011
<MerchandiseERPReplicationRequestMessage> <Merchandise> <MaterialMerchandiseTypeCode>	String			Item Type Code	itemTypeCode 1 → 1 2 → 2 3 → 3 4 → 4	

Example

```

<?xml version="1.0" encoding="utf-8"?>
<n0:MerchandiseERPReplicationBulkRequest xmlns:n0="http://sap.com/xi/SAPGlobal20/Global"
xmlns:prx="urn:sap.com:proxy:GE9:/1SAI/TASE09325101CF7B23E85D7:750">
  <MessageHeader>
    <ID>005056BD41161ED8BCC5B245C73D00C0</ID>
    <CreationDateTime>2018-11-27T10:16:58Z</CreationDateTime>
    <SenderBusinessSystemID>BS_GE9CLNT542</SenderBusinessSystemID>
    <RecipientBusinessSystemID>INTEGRATION_SERVER_GX1</RecipientBusinessSystemID>
  </MessageHeader>
  <MerchandiseERPReplicationRequestMessage>
    <MessageHeader>
      <ID>005056BD41161ED8BCC5B245C73D00C0</ID>
      <CreationDateTime>2018-11-27T10:16:58Z</CreationDateTime>
      <SenderBusinessSystemID>BS_GE9CLNT542</SenderBusinessSystemID>
      <RecipientBusinessSystemID>INTEGRATION_SERVER_GX1</RecipientBusinessSystemID>
    </MessageHeader>
    <Merchandise>
      <InternalID>16</InternalID>
      <MaterialTypeCode>HAW1</MaterialTypeCode>
      <MaterialMerchandiseTypeCode>1</MaterialMerchandiseTypeCode>
      <ReceivingStore>
        <StoreInternalID>1025</StoreInternalID>
      </ReceivingStore>
      <PointOfSaleProcessingCondition>
        <PriceRequiredIndicator>>false</PriceRequiredIndicator>
        <DiscountAllowedIndicator>>false</DiscountAllowedIndicator>
        <WeightingForPricingRequiredIndicator>>false</WeightingForPricingRequiredIndicator>
        <TaxIncludedIndicator>>false</TaxIncludedIndicator>
      </PointOfSaleProcessingCondition>
      <TextOnPointOfSaleRegisterVisibleIndicator>>true</TextOnPointOfSaleRegisterVisibleIndicator>
      <Description>
        <Description languageCode="de">Schreibmappe</Description>
      </Description>
      <GlobalTradeItemNumber>
        <ProductStandardID>2050000000102</ProductStandardID>
        <ProductStandardIDType>IE</ProductStandardIDType>
        <MeasureUnitCode>PCE</MeasureUnitCode>
        <ProductStandardMainIndicator>>true</ProductStandardMainIndicator>
      </GlobalTradeItemNumber>
      <QuantityUnit>
        <MeasureUnitCode>PCE</MeasureUnitCode>
        <BaseQuantityUnitIndicator>>true</BaseQuantityUnitIndicator>
        <SupplementaryQuantityUnitUsageCode>5</SupplementaryQuantityUnitUsageCode>
      </QuantityUnit>
      <QuantityConversion>
        <Quantity unitCode="PCE">1.0</Quantity>
        <CorrespondingQuantity unitCode="PCE">1.0</CorrespondingQuantity>
      </QuantityConversion>
      <ProductCategory>
        <InternalID>1025</InternalID>
        <ProductCategoryHierarchyID>1</ProductCategoryHierarchyID>
        <ProductCategoryHierarchyTypeCode>3</ProductCategoryHierarchyTypeCode>
      </ProductCategory>
      <MeasureUnitCodeSpecificDescription>
        <MeasureUnitCode>PCE</MeasureUnitCode>
        <Description languageCode="de">Schreibmappe Bontext</Description>
      </MeasureUnitCodeSpecificDescription>
      <TaxClassification>
        <CountryCode>DE</CountryCode>
        <TaxTypeCode>1</TaxTypeCode>
        <TaxRateType>1</TaxRateType>
        <Percent>19.0</Percent>
        <ProductTaxationCharacteristicsCode>A1</ProductTaxationCharacteristicsCode>
      </TaxClassification>
      <ProcurementPriceInformation>
        <ProductStandardID>2050000000102</ProductStandardID>
        <OrderMeasureUnitCode>PCE</OrderMeasureUnitCode>
        <OrderTransactionCurrencyCode>EUR</OrderTransactionCurrencyCode>
        <StrategicPurchasingFunctionalUnitID>1025</StrategicPurchasingFunctionalUnitID>
        <SupplierInternalID>0000100005</SupplierInternalID>
        <ValidityPeriod>
          <StartDate>2018-11-27</StartDate>
          <EndDate>2021-08-21</EndDate>
        </ValidityPeriod>
        <PriceSpecification>

```

```

        <PriceSpecificationElementTypeCode>PB00</PriceSpecificationElementTypeCode>
        <PriceSpecificationElementCategoryCode>1</PriceSpecificationElementCategoryCode>
        <ValidityPeriod>
            <IntervalBoundaryTypeCode>3</IntervalBoundaryTypeCode>
            <StartTimePoint>
                <TypeCode></TypeCode>
                <Date>2018-11-27</Date>
            </StartTimePoint>
            <EndTimePoint>
                <TypeCode></TypeCode>
                <Date>9999-12-31</Date>
            </EndTimePoint>
        </ValidityPeriod>
        <Amount currencyCode="EUR">2.84</Amount>
        <BaseQuantity unitCode="PCE">1.0</BaseQuantity>
        <OrderPriceAmount currencyCode="EUR">2.84</OrderPriceAmount>
        <OrderBaseQuantity unitCode="PCE">1.0</OrderBaseQuantity>
    </PriceSpecification>
</ProcurementPriceInformation>
<SalesPriceInformation>
    <ProductStandardID>2050000000102</ProductStandardID>
    <MeasureUnitCode>PCE</MeasureUnitCode>
    <SalesOrganisationID>1025</SalesOrganisationID>
    <DistributionChannelCode>V1</DistributionChannelCode>
    <ValidityPeriod>
        <StartDate>2018-11-27</StartDate>
        <EndDate>9999-12-31</EndDate>
    </ValidityPeriod>
    <PriceSpecification>
        <PriceSpecificationElementTypeCode>VKP0</PriceSpecificationElementTypeCode>
        <PriceSpecificationElementCategoryCode>1</PriceSpecificationElementCategoryCode>
        <Amount currencyCode="EUR">5.99</Amount>
        <BaseQuantity unitCode="PCE">1.0</BaseQuantity>
    </PriceSpecification>
</SalesPriceInformation>
</Merchandise>
</MerchandiseERPReplicationRequestMessage>
<MerchandiseERPReplicationRequestMessage>
    ...
</MerchandiseERPReplicationRequestMessage>
</n0:MerchandiseERPReplicationBulkRequest>

```

Code Block 20 Item und Item Categories

Category and CategoryHierarchy

All values are discarded before processing their excess spaces.

ERP field	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code
<ProductCategoryHierarchy> <MerchandiseCategory> <Description> <Description languageCode="xyz">	String	TK-Ware2		category "Name"	Only the items with the LanguageCode indicated in the Configuration file	
<ProductCategoryHierarchy> <MerchandiseCategory> <InternalID>	String	1025	x	category ID		<ul style="list-style-type: none"> not set: 51011 category does already exist: 51029
<ProductCategoryHierarchy> <MerchandiseCategory> <MerchandiseCategoryHierarchyRelationship> <PredecessorID>	String	1025		parent category ID	The following applies to root categories: InternalID = PredecessorID. These root categories are not applied in the category hierarchies table of our database. However, they exist as usual in the categories table.	

Example

```

<?xml version="1.0" encoding="utf-8"?>
<n0:ProductCategoryHierarchyERPReplicationRequest xmlns:n0="http://sap.com/xi/EA-RETAIL/Global2"
xmlns:prx="urn:sap.com:proxy:GE9:/1SAI/TAS0BA4DA6B02A34BE3E1E9:750">
  <MessageHeader>
    <ID>005056BD41161ED8BCC5E67D568360C0</ID>
    <CreationDateTime>2018-11-27T10:28:41Z</CreationDateTime>
    <SenderBusinessSystemID>BS_GE9CLNT542</SenderBusinessSystemID>
    <RecipientBusinessSystemID>INTEGRATION_SERVER_GX1</RecipientBusinessSystemID>
  </MessageHeader>
  <ProductCategoryHierarchy Actioncode="04">
    <MerchandiseCategory>
      <AssignmentAllowedIndicator>true</AssignmentAllowedIndicator>
      <InternalID>1025</InternalID>
      <Property>
        <Id>FARBE01</Id>
        <PreferredName languageCode="de">Farbe01</PreferredName>
        <InheritedIndicator>true</InheritedIndicator>
        <AllowedValue>
          <IntervalBoundaryTypeCode>
            </IntervalBoundaryTypeCode>
          <ObjectPropertyValue>
            <Name>BLAU</Name>
            <Text languageCode="de">Blau</Text>
          </ObjectPropertyValue>
        </AllowedValue>
        <AllowedValue>
          <IntervalBoundaryTypeCode>
            </IntervalBoundaryTypeCode>
          <ObjectPropertyValue>
            <Name>ROSE</Name>
            <Text languageCode="de">Blau</Text>
          </ObjectPropertyValue>
        </AllowedValue>
        <AllowedValue>
          <IntervalBoundaryTypeCode>
            </IntervalBoundaryTypeCode>
          <ObjectPropertyValue>
            <Name>ROT</Name>
            <Text languageCode="de">Blau</Text>
          </ObjectPropertyValue>
        </AllowedValue>
        <AllowedValue>
          <IntervalBoundaryTypeCode>
            </IntervalBoundaryTypeCode>
          <ObjectPropertyValue>
            <Name>SCHWARZ</Name>
            <Text languageCode="de">Blau</Text>
          </ObjectPropertyValue>
        </AllowedValue>
        <AllowedValue>
          <IntervalBoundaryTypeCode>
            </IntervalBoundaryTypeCode>
          <ObjectPropertyValue>
            <Name>WEISS</Name>
            <Text languageCode="de">Blau</Text>
          </ObjectPropertyValue>
        </AllowedValue>
      </Property>
      <Property>
        <Id>GROESSE01</Id>
        <PreferredName languageCode="de">Grösse 01</PreferredName>
        <InheritedIndicator>true</InheritedIndicator>
        <AllowedValue>
          <IntervalBoundaryTypeCode>
            </IntervalBoundaryTypeCode>
          <ObjectPropertyValue>
            <Name>L</Name>
            <Text languageCode="de">L</Text>
          </ObjectPropertyValue>
        </AllowedValue>
        <AllowedValue>
          <IntervalBoundaryTypeCode>
            </IntervalBoundaryTypeCode>
          <ObjectPropertyValue>
            <Name>M</Name>
            <Text languageCode="de">L</Text>
          </ObjectPropertyValue>
        </AllowedValue>
      </Property>
    </MerchandiseCategory>
  </ProductCategoryHierarchy>
</n0:ProductCategoryHierarchyERPReplicationRequest>

```



```

        </ObjectPropertyValue>
    </AllowedValue>
    <AllowedValue>
        <IntervalBoundaryTypeCode>
        </IntervalBoundaryTypeCode>
        <ObjectPropertyValue>
            <Name>S</Name>
            <Text languageCode="de">L</Text>
        </ObjectPropertyValue>
    </AllowedValue>
    <AllowedValue>
        <IntervalBoundaryTypeCode>
        </IntervalBoundaryTypeCode>
        <ObjectPropertyValue>
            <Name>XL</Name>
            <Text languageCode="de">L</Text>
        </ObjectPropertyValue>
    </AllowedValue>
    <AllowedValue>
        <IntervalBoundaryTypeCode>
        </IntervalBoundaryTypeCode>
        <ObjectPropertyValue>
            <Name>XS</Name>
            <Text languageCode="de">L</Text>
        </ObjectPropertyValue>
    </AllowedValue>
</Property>
<MerchandiseCategoryHierarchyRelationship>
    <PredecessorID>1025</PredecessorID>
</MerchandiseCategoryHierarchyRelationship>
<Description>
    <Description languageCode="de">TK-Ware2</Description>
</Description>
</MerchandiseCategory>
</ProductCategoryHierarchy>
</n0:ProductCategoryHierarchyERPReplicationRequest>

```

Code Block 21 Categories and category hierarchies

Configuration

General

Tag	Format	Example	Mandatory	Default	Description
languageCode	String	en		First language code is used.	Only the items and categories with the indicated Language Code are imported. You can configure more than one code, then separate the codes by comma: de,en If you do not provide the tag or the tag's value is empty, then the first language code that is provided is used.
importMode	INITIAL UPDATE	UPDATE		UPDATE	INITIAL = Deletes masterdata before the import starts: Items, item categories, categories, category hierarchies. UPDATE = Updates existing entities and adds new entities.
<hierarchyNameMappings> <hierarchyNameMapping> <id>1</id> <name>Standard</name> </hierarchyNameMapping> </hierarchyNameMappings>					Configure names for hierarchy IDs. In the XML format you can just transport hierarchy IDs. If you want to configure reasonable names that are displayed as taxonomy names in the UI, you can define a mapping for each id. Unmapped ids are displayed as they are in the UI.

Future Prices

Tag	Format	Example	Mandatory	Default	Description
infinityDate	String dd.MM.yyyy	31.12.9999		31.12.9999	This date expresses infinity. If this date is set for a future price, then valid to is unbounded
nowOffset	Integer >= 0	0		1	Future begins from today + this offset. This is important for deciding to create a future price or not.
futurePriceImportEnabled	true, false	true		false	true = If future prices are found, import them false = If future prices are found, do NOT import them. However, import the first price that is found to be in valid from/to interval for the time of the import.

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

General

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

XML Reading

Please see details about the configuration here.

Example Categories

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <importMode>UPDATE</importMode>
  <languageCode></languageCode>
  <xmlReader>
    <charset>UTF-8</charset>
  </xmlReader>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>Category</mappingId>
      <regExpression>ProductCategoryHierarchyERPReplicationRequest.xml</regExpression>
      <type>Category</type>
      <defaultName>ProductCategoryHierarchyERPReplicationRequest.xml</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>CategoryHierarchy</mappingId>
      <regExpression>ProductCategoryHierarchyERPReplicationRequest.xml</regExpression>
      <type>CategoryHierarchy</type>
      <defaultName>ProductCategoryHierarchyERPReplicationRequest.xml</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

Code Block 22 Sample configuration

Example Items

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <importMode>UPDATE</importMode>
  <languageCode></languageCode>
  <xmlReader>
    <charset>UTF-8</charset>
  </xmlReader>
  <infinityDate>31.12.2099</infinityDate>
  <nowOffset>0</nowOffset>
  <futurePriceImportEnabled>>false</futurePriceImportEnabled>  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>Item</mappingId>
      <regExpression>MerchandiseERPReplicationBulkRequest.xml</regExpression>
      <type>Item</type>
      <defaultName>MerchandiseERPReplicationBulkRequest.xml</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>ItemCategory</mappingId>
      <regExpression>MerchandiseERPReplicationBulkRequest.xml</regExpression>
      <type>ItemCategory</type>
      <defaultName>MerchandiseERPReplicationBulkRequest.xml</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

Code Block 23 Sample configuration

2.5.10.4 Item Attribute Import - SAP IDOC ARTMAS

Summary

This import imports item attributes from SAP (ERP, ECC, S/4HANA) using the **ARTMAS09 IDoc in XML format**.

An ARTMAS record matches existing items via the material id. If this id is not set in the item data, the importer is not able to update any item.

■ The service supports only the update mode. So you need to import items before importing attributes.

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET	Previous deletion of the current data	Ad hoc reading of the files from the file system or by an HTTP endpoint
POST	Previous deletion of the current data	The files in the ZIP archive are transferred
Scheduler	Previous deletion of the current data	Time-controlled reading of the files from the file system or from an HTTP endpoint

Mapping and Validation

To update an existing item with new attribute data, we need to reference that item. This is done by the material ID. The import reads the material ID for a record and then looks up the corresponding item ID in the database.

Therefore it requires that the item import imported items with **materialId** field set.

XML field	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code
<IDOC> <E1BPE1MATHEAD> <MATERIAL>	String	MR12333	x	Material ID	Find an item with a matching materialId and update that item.	<ul style="list-style-type: none">No matching item for Material ID found: 151054

Example:

```
<?xml version="1.0"?>
<ARTMAS09>
  <IDOC BEGIN="1">
    <EDI_DC40 SEGMENT="1">
      <TABNAM><![CDATA[EDI_DC40]]></TABNAM>
      <MANDT>400</MANDT>
      <DOCNUM>0000000001581006</DOCNUM>
      <DOCREL>753</DOCREL>
      <STATUS>03</STATUS>
      <DIRECT>1</DIRECT>
      <OUTMOD>2</OUTMOD>
      <IDOC TYP>ARTMAS09</IDOC TYP>
      <MESTYP>ARTMAS</MESTYP>
      <STDMES>ARTMAS</STDMES>
      <SNDPDR><![CDATA[SAPS4Q_001]]></SNDPDR>
      <SNDPRT>LS</SNDPRT>
      <SNDPRN><![CDATA[GKO_POS]]></SNDPRN>
      <RCVPDR><![CDATA[PIQ_00_001]]></RCVPDR>
      <RCVPRT>LS</RCVPRT>
      <RCVPRN><![CDATA[GKO_POS]]></RCVPRN>
      <CREDAT>20190111</CREDAT>
      <CRETIM>163702</CRETIM>
      <SERIAL>20180514102220</SERIAL>
    </EDI_DC40>
    <E1BPE1MATHEAD SEGMENT="1">
      <MATERIAL>MR550270</MATERIAL>
      <MATL_TYPE>ZFRT</MATL_TYPE>
      <MATL_GROUP>RF17111</MATL_GROUP>
      <MATL_CAT>00</MATL_CAT>
      <BASIC_VIEW>X</BASIC_VIEW>
      <LIST_VIEW>X</LIST_VIEW>
      <SALES_VIEW>X</SALES_VIEW>
      <LOGDC_VIEW>X</LOGDC_VIEW>
      <LOGST_VIEW>X</LOGST_VIEW>
      <POS_VIEW>X</POS_VIEW>
      <MATERIAL_LONG>MR550270</MATERIAL_LONG>
    </E1BPE1MATHEAD>
  </IDOC>
</ARTMAS09>
```

You then can configure any ARTMAS field to import and update any item field (also extension columns are supported) using `attributeFieldMappings` in the XML configuration file.

Example:

```
<attributeFieldMappings>
  <attributeFieldMapping>
    <sourcePath>ARTMAS09.IDOC.E1BPE1MARARTX.BRAND_ID</sourcePath>
    <destinationField>brand</destinationField>
    <dataType>character</dataType>
    <isMandatory>true</isMandatory>
  </attributeFieldMapping>
</attributeFieldMappings>
```

Tag	Format	Mandatory	Default	Example	Description
sourcePath	String separated by "."	x		ARTMAS09.IDOC.E1BPE1MARARTX.BRAND_ID	The path of tags in the XML file to the field containing the value to map to the destinationField.
destinationField	String	x		brand	Column name of the item. For product standard fields, please take a look at the Masterdata CSV importer about the supported field names. You can also import values to extension fields.
dataType	character numeric timestamp	x		character	Datatype of the field. The importer then validates the data against this type.
isMandatory	true, false	x		true	false: true: Validates if the field is set, i.e. that it is not empty or does not exist at all. If it is not set, the error 151057 is written to the protocol and the item is discarded.

Configuration

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

For general description, see "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

XML Reading

Please see details about the configuration here.

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>Attributes</mappingId>
      <regExpression>.*artmas.*xml</regExpression>
      <type>Stock</type>
      <defaultName>artmas.xml</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <xmlReader>
    <charset>UTF-8</charset>
  </xmlReader>
  <attributeFieldMappings>
    <attributeFieldMapping>
      <sourcePath>ARTMAS09.IDOC.E1BPE1MARARTX.BRAND_ID</sourcePath>
      <destinationField>brand</destinationField>
      <dataType>character</dataType>
      <isMandatory>true</isMandatory>
    </attributeFieldMapping>
  </attributeFieldMappings>
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

2.5.10.5 Item Stock Import - SAP InventoryBulkRequest_Out

Summary

This import imports item attributes from SAP (ERP, ECC, S/4HANA) using the InventoryBulkRequest_Out Service **in XML format**.

The stock importer imports the stock in the base quantity unit of measure. For each record the corresponding item is matched via the material id. If the item exists, we check if the base quantity unit of measure of the item matches the one of the stock update record. If not, it is discarded.

■ The service supports only the update mode. So you need to import items before importing attributes.

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET	Previous deletion of the current data	Ad hoc reading of the files from the file system or by an HTTP endpoint
POST	Previous deletion of the current data	The files in the ZIP archive are transferred
Scheduler	Previous deletion of the current data	Time-controlled reading of the files from the file system or from an HTTP endpoint

Mapping and Validation

XML field	Format	Example	Mandatory Field	Description	Mapping	Validation and Error Code
<Location><InternalID>	String	DSC1		ID of the location	Not mapped. We report this ID as event in the import protocol.	
<Location><TypeCode>	String	1040			Not mapped. We report this typecode as event in the import protocol.	
<Item><Material><InternalID>	String	17	x	Material ID of the item	Find a matching item that has this material ID	<ul style="list-style-type: none"> If there is no matching item with this material ID: 151054 Unit of Measure not set: 151056
<Item><Quantity><InventoryQuantity unitCode=xxx>	Value: Float unitCode: String	Value: 10.0 unitCode: PCE	x	Measurement unit code and the stock value.	Unit code must match the item's unit code. Then set the stock value to item.stock	<ul style="list-style-type: none"> If UOMs do not match for this item: 151055

Example

```
<?xml version="1.0" encoding="utf-8"?>
<n0:InventoryBulkRequest xmlns:n0="http://sap.com/xi/APPL/Global2"
  xmlns:prx="urn:sap.com:proxy:QI5:/1SAI/TAS8AE88EB00DE10E44AB9C:750">
  <MessageHeader>
    <ID>FA163E7B192D1ED9AFD1685FE0D703E2</ID>
    <CreationDateTime>2019-08-14T00:00:00Z</CreationDateTime>
    <SenderBusinessSystemID>QI5_410</SenderBusinessSystemID>
    <RecipientBusinessSystemID>Q5R_410</RecipientBusinessSystemID>
  </MessageHeader>
  <InventoryRequestMessage>
    <Inventory>
      <Location>
        <InternalID>DSC1</InternalID>
        <TypeCode>1040</TypeCode>
      </Location>
      <Item>
        <InventoryUsabilityCode>1</InventoryUsabilityCode>
        <Material>
          <InternalID>17</InternalID>
        </Material>
        <Quantity>
          <InventoryQuantity unitCode="PCE">123</InventoryQuantity>
        </Quantity>
      </Item>
    </Inventory>
  </InventoryRequestMessage>
</n0:InventoryBulkRequest>
```

Configuration

General

```
<locationId>R103</locationId>
<locationTypeCode>1040</locationTypeCode>
```

Tag	Format	Mandatory	Default	Example	Description
locationId	String			R103	Include only inventory from this locationId
locationTypeCode	String			1040	Include only inventory with this type code

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

For general description, see "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

XML Reading

Please see details about the configuration here.

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>Stock</mappingId>
      <regExpression>stock.*xml</regExpression>
      <type>Stock</type>
      <defaultName>stock.xml</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <xmlReader>
    <charset>UTF-8</charset>
  </xmlReader>
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

2.5.10.6 Item Import - Prices CSV

Summary

This import is used to import prices in the CSV format.

You are able to set one or more future prices for an item. The import supports both fixed and relative price values.

Name for Selection

"Standard - Manual Prices CSV"

Update Strategy

Import Mode	Description
UPDATE	For each record the importer looks for an existing record that matches the item id (pid). If there is no existing record for this ID and error is reported for that item. If the item exists, an additional change record is added for that item. Existing changes are not touched as long as there are no conflicts. In conflict case the new change wins.
INITIAL	For each record the importer looks for an existing record that matches the item id (pid). If there is no existing record for this ID and error is reported for that item. If the item exists, all changes are deleted and the new ones are imported.

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET	Item update	Ad hoc reading of the file from the file system or from an HTTP endpoint
POST	Item update	The file is transferred as ZIP archive
Scheduler	Item update	Time-controlled reading of the file from the file system or by an HTTP endpoint

Example

```
pid|changeType|validFrom|validTo|fixedPrice|relativeValue|relativeMeasure|relativeReferenceField|calculatedWhen|minLimit|maxLimit|expiryDate
000019595|FP|2021-06-01|2021-06-10|15.00| || || |
000019595|FP|2021-08-01| ||16.00| || || |
000019613|RP|2021-06-01|2021-06-10| |-10%|regularUnitPrice|NOW| || |
000019613|RP|2022-06-01|2022-06-10| |10|MONEY|regularUnitPrice|VALIDFROM| || |
000019613|MPL| |2022-06-10| || ||10.00|12.00|
000019613|SD| |2022-06-10| || || ||2023-01-01
```


CSV field	Format	Example	Mandatory	Description	Default if empty	Validation and Error Code
pid	String	112344	x	Item ID	Error	<ul style="list-style-type: none"> not set: 51011
changeType	Fixed values: <ul style="list-style-type: none"> FP RP MPL MUP SD 	FP	x	<ul style="list-style-type: none"> FP: Fixed Price RP: Relative Price MPL: Min/Max Pricing Limit MUP: Min/Max Unit Price SD 		<ul style="list-style-type: none"> not set: 51011
validFrom	Date: yyyy-MM-dd	2021-02-01		Change is valid from this day on	Date of import	
validTo	Date: yyyy-MM-dd	2021-02-01		Change is valid up to this day	unlimited	
fixedPrice	Number in US format, no thousands separator, "." for comma	12.33	x for changeType=FP	Set a fixed value for the price		<ul style="list-style-type: none"> not set: 51011
relativeValue	Number in US format, no thousands separator, "." for comma also negative values are supported	12.00 -3.11	x for changeType=RP	Reduce/increase the relativeReferenceField by this value. Value is interpreted according to relativeMeasure.		<ul style="list-style-type: none"> not set: 51011
relativeMeasure	Fixed values: <ul style="list-style-type: none"> % MONEY 	%	x for changeType=RP	<ul style="list-style-type: none"> %: interpret relativeValue as percentage value MONEY: interpret relativeValue as money value 		<ul style="list-style-type: none"> not set: 51011
relativeReferenceField	String	regularUnitPrice		Field that is used to compute the price value. Default is regularUnitPrice. For a list of fields see the masterdata CSV importer. Besides that extension fields are supported as well (x...).	regularUnitPrice	
calculatedWhen	Fixed values: <ul style="list-style-type: none"> NOW VALIDFROM 	NOW		<ul style="list-style-type: none"> NOW: Calculate the price based on the value of the relativeReferenceField that is valid now VALIDFROM: Calculate the price based on the value of the relativeReferenceField at the validFrom time 		
minLimit	Number in US format, no thousands separator, "." for comma	10.00	x (at least minLimit or maxLimit) for changeType=MPL/MUP	Minimum border for the price. Can be a minimum price limit or minimum unit price, depending on the change type.		
maxLimit	Number in US format, no thousands separator, "." for comma	12.00	x (at least minLimit or maxLimit) for changeType=MPL/MUP	Minimum border for the price. Can be a minimum price limit or minimum unit price, depending on the change type.		
expiryDate	Date: yyyy-MM-dd	2021-02-01	x for changeType=SD	Sale expiry Date		

Time Range Validation

If you add more rules for one single item in one import, make sure that the rules do not overlap. Otherwise the records are discarded.

Example:

1. Fixed price for item A from today on
2. Relative price for item A from today on

This is a conflict and both items are discarded.

Another example that does not work:

1. Fixed price for item A from today on
2. Relative price for item A from next week on

There is no conflict handling for records for one import. There is a conflict handling for the update mode for consecutive imports that automatically ends the old rules.

Configuration

General

```
<roundingMode>CASHROUNDING</roundingMode>  
<importMode>INITIAL</importMode>
```

Tag	Format	Mandatory	Default	Example	Description
roundingMode	CASHROUNDING, GLOBALRULES		CASHROUNDING	GLOBALRULES	<ul style="list-style-type: none">• CASHROUNDING: Takes into account the business unit currency in cash rounding mode• GLOBALRULES: Takes into account the global rounding rules of the business unit
importMode	INITIAL UPDATE		INITIAL		See Chapter "Update Strategy"

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

General

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

CSV Reading

Please see details about the configuration here.

Data Mapping

This import supports mapping of CSV data fields to internal data fields and fixed values via XML configuration.

This import supports mapping of fields of those entities:

Entity	<type>
Manual Overrides	ManualOverride

For more details about the mapping, please refer to the documentation here.

Mapping Example

```
<fieldMappings>
  <fieldMapping>
    <type>ManualOverride</type>
    <mode>ColumnName</mode>
    <sourceField>myMinLimit</sourceField>
    <destinationField>minLimit</destinationField>
  </fieldMapping>
</fieldMappings>
```

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>ManualOverride</mappingId>
      <regExpression>manualprices.*.csv</regExpression>
      <type>ManualOverride</type>
      <defaultName>manualprices.csv</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <csvReader>
    <charset>UTF-8</charset>
    <readHeader>true</readHeader>
    <escapeMode>BACKSLASH</escapeMode>
    <columnSeparator>|</columnSeparator>
  </csvReader>
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

2.5.10.7 Merchandise Group Import - SAP IDOC WPDWGR01

Summary

This import is used to import categories and category hierarchy data from SAP (ERP, ECC, S/4HANA) using the **WPDWGR01 IDoc in XML format**.

Communication

The following options are supported for the communication:

Trigger	Description
GET	Ad hoc reading data from a source (e.g. filesystem, FTP, HTTP...)
POST	Put data to the interface and start processing
Scheduler	Time-controlled reading of the files from a source (e.g. filesystem, FTP, HTTP...)

Update Strategy

Strategy	Description
INITIAL	Before importing data current item data is out dated by setting valid to date to the time stamp before the import. In this way the data is still in the database for historical data access, but not visible for current accesses.
UPDATE	The new data is added. Existing data is updated.

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<WPDWGR01>
  <IDOC BEGIN="1">
    <EDI_DC40 SEGMENT="1">
      <TABNAM>EDI_DC40</TABNAM>
      <MANDT>003</MANDT>
      <DOCNUM>0000000547027841</DOCNUM>
      <DOCREL>740</DOCREL>
      <STATUS>30</STATUS>
      <DIRECT>1</DIRECT>
      <OUTMOD>2</OUTMOD>
      <IDOCTYP>WPDWGR01</IDOCTYP>
      <MESTYP>WPDWGR</MESTYP>
      <SNDPOR>SAPRP1</SNDPOR>
      <SNDPRT>LS</SNDPRT>
      <SNDPRN>RP1CLNT003</SNDPRN>
      <RCVPOR>OP1CLNT003</RCVPOR>
      <RCVPRT>LS</RCVPRT>
      <RCVPRN>OP1CLNT003</RCVPRN>
      <CREDAT>20190716</CREDAT>
      <CRETIM>103043</CRETIM>
      <SERIAL>20190716103042</SERIAL>
    </EDI_DC40>
    <E1WPW01 SEGMENT="1">
      <FILIALE>0000002884</FILIALE>
      <AENDKENNZ>MODI</AENDKENNZ>
      <AKTIVDATUM>20190716</AKTIVDATUM>
      <AENDDATUM>00000000</AENDDATUM>
      <WARENGR>1000</WARENGR>
      <E1WPW02 SEGMENT="1">
        <BEZEICH>Fisch</BEZEICH>
        <HIERARCHIE>00</HIERARCHIE>
        <VERKNUEPFG>10100</VERKNUEPFG>
      </E1WPW02>
    </E1WPW01>
  </IDOC>
</WPDWGR01>
```

Mapping and Validation

Segment	Field	Format	Example	Mandatory	Description	Validation	Error Code
<E1WPW01 SEGMENT="1">	<WARENGR>	String	1000	x	Merchandise group ID		<ul style="list-style-type: none"> not set: 51011 item does already exist: 51029
	<AENDKENNZ>	String	MODI		Change flag	If this tag is present, only records with "MODI" are processed. All others with a non empty value are skipped.	
<E1WPW02 SEGMENT="1">	<BEZEICH>	String	Fisch		Name of the category		
	<VERKNUEPFG>	String	10100		Parent merchandise group ID	If set, then this group is a child of this parent.	

Configuration

General

```
<importMode>INITIAL</importMode><defaultTaxonomy>MyTaxonomy</defaultTaxonomy>
```

Tag	Format	Mandatory	Default	Example	Description
defaultTaxonomy	String		Standard	MyTaxonomy	If you want to import different category hierarchies, you can use a different name. This name is the taxonomy.
importMode	INITIAL UPDATE		UPDATE		See Chapter "Update Strategy"

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

General

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

XML Reading

Please see details about the configuration here.

2.5.10.8 Master Data Import - GK/Retail XML

Summary

This import is used to import master data in XML format from the GK/Retail Store Device Control (SDC). There are two importers to import:

1. Item data and category assignment
2. Merchandise groups and hierarchy

Communication

The following options are supported for the communication:

Trigger	Description
GET	Ad hoc reading data from a source (e.g. filesystem, FTP, HTTP...)
POST	Put data to the interface and start processing
Scheduler	Time-controlled reading of the files from a source (e.g. filesystem, FTP, HTTP...)

Update Strategy

Strategy	Description
INITIAL	Before importing data current item data is out dated by setting valid to date to the time stamp before the import. In this way the data is still in the database for historical data access, but not visible for current accesses.
UPDATE	The new data is added. Existing data is updated.

Category and Category Hierarchy

Example

```
<?xml version='1.0' encoding='UTF-8'?>
<MerchandiseHierarchyGroupList xmlns:data-extension-map="http://www.gk-
software.com/schema/core/server/extension-map/map/map-1.0" xmlns:importDomain="http://www.gk-
software.com/storeweaver/master_data/import_domain/2.4.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns="http://www.gk-software.com/storeweaver/master_data/mhg/3.5.0" NumberOfMHG="2195"
xsi:schemaLocation="http://www.gk-software.com/storeweaver/master_data/mhg/3.5.0
masterData_MerchandiseHierarchyGroup.xsd">
  <MerchandiseHierarchyGroup ChangeType="MODIFY">
    <MerchandiseHierarchyGroupID>1</MerchandiseHierarchyGroupID>
    <Name>Warengruppe 1</Name>
    <Description>Warengruppe 1</Description>
    <ExternalPackageID>0000000092212757</ExternalPackageID>
    <LevelName>00</LevelName>
  </MerchandiseHierarchyGroup>
  <MerchandiseHierarchyGroup ChangeType="MODIFY">
    <MerchandiseHierarchyGroupID>2</MerchandiseHierarchyGroupID>
    <Name>Warengruppe 2</Name>
    <Description>Warengruppe 2</Description>
    <ParentID>1</ParentID>
    <ExternalPackageID>0000000092212757</ExternalPackageID>
    <LevelName>01</LevelName>
  </MerchandiseHierarchyGroup>
</MerchandiseHierarchyGroupList>
```

Mapping

Field	Format	Example	Mandatory	Description	Validation	Error Code
<MerchandiseHierarchyGroupID>	String	1000	x	Merchandise group ID		<ul style="list-style-type: none">not set: 51011item does already exist: 51029
<Name>	String	Fisch		Name of the category		
<ParentID>	String	10100		Parent merchandise group ID	If set, then this group is a child of this parent.	

Item and Category Assignment

Mapping Details

Master/Variants

The MainPOSItemID is mapped to the Material ID in any case.

An item can exist in more than one unit of measure, so there can be more UOMItem entries for each item.

If there are more UOMItem entries the MainPOSItemID is also set as Master ID for each UOMItem which is then treated as a variant.

Unit of Measures and Quantity

If conversionDenominator and conversionNumerator tags are both filled with a value then:

quantity = conversionNumerator/conversionDenominator

In other cases quantity is set to 1

Both UOM and base UOM are supported.

Mapping of Item Texts to Item Name

The name of the item is put together from all <Text> tags in <UOMItemText>.

First the entries are filtered:

- By language ID, if you configured a special language code in the configuration.
- The resulting text entries are filtered by the text classes to match your configuration

- Finally those texts with text numbers mentioned in the configuration for `excludedTextNumbers` are discarded

The resulting texts are concatenated, each one separated by a blank space character.

Handling Future Prices

There is a configuration switch to enable the import of future prices: **`futurePriceImportEnabled`**.

Future Prices import disabled

In case the import is disabled, no future prices are imported. However, also prices that do not have a valid from/to (effective, expiration) date information in the price list, won't be imported.

To import a price the date of the import must be in a valid from/to interval of the price definition.

Examples:

- Settings:
 - Import Date = 08.07.2021
 - `futurePriceImportEnabled` = false
- Example 1:
 - EffectiveDate = 03.01.2020
 - ExpiryDate = 05.01.2020
 - **No price is imported**
- Example 2:
 - EffectiveDate = 03.01.2020
 - ExpiryDate = 05.01.2022
 - **Price is imported**

Future Prices import enabled

If future prices are enabled you can fine tune, if a price is imported as a future price or not and if the valid to boundary is open or not.

Valid to Boundary

Future prices can be stored as:

- Valid to a concrete date
- Valid to forever

To distinguish between these two cases, the importer needs to know what "forever" means. You can configure the concrete date for it in the configuration parameter **`infinityDate`**.

Examples:

- Settings:
 - `infinityDate` = 31.12.2099
 - `futurePriceImportEnabled` = true
- Example 1:
 - EffectiveDate = 03.01.2020
 - ExpiryDate = 31.12.2025
 - **Valid to a concrete date: 30.12.2025**
- Example 2:
 - EffectiveDate = 03.01.2020
 - ExpiryDate = 31.12.2099

- **Validity is unbounded**

When does the future begin?

You might want to import future prices, but maybe not all prices are future prices that just start tomorrow.

For instance you import the master data on 08.07. and the EffectiveDate is 09.07.21. In this case the importer creates a future price record. If you don't want that, you can tell the importer with the parameter **nowOffset** to move the start of the future to the next day.

- Settings:
 - infinityDate = 31.12.2099
 - futurePriceImportEnabled = true
 - nowOffset = 1
- Example 1:
 - Date of import = 08.07.2021
 - EffectiveDate = 09.07.2021
 - ExpiryDate = 31.12.2099
 - 1 day is added to the day of import, so 09.07.2021 is treated as now. For any prices that are getting effective now to infinity (see infinityDate parameter to control that), **no future price is created but just the price is set as regular price.**
- Example 2:
- Date of import = 08.07.2021
- EffectiveDate = 09.07.2021
- ExpiryDate = 31.12.2022
- 1 day is added to the day of import, so 09.07.2021 is treated as now. However, as the expiry date is not infinity, **a future price record is created with a valid to 31.12.2022.**

Example

```

<?xml version='1.0' encoding='utf-8'?>
<ItemList xmlns='http://www.gk-software.com/storeweaver/master_data/item/3.13.0' NumberOfItems='172'>
  <Item ChangeType=''>
    <BusinessUnitAssignment>
      <ExternalBusinessUnitID>0000002340</ExternalBusinessUnitID>
    </BusinessUnitAssignment>
    <ItemID>000000000000260606</ItemID>
    <BonText>Winston Black BP 4XL 34er 10.00</BonText>
    <Description>Winston Black BP 4XL 34er 10.00</Description>
    <TaxGroupID>A1</TaxGroupID>
    <BaseUOMCode>PCE</BaseUOMCode>
    <MinimumShelfLifeDayCount>0</MinimumShelfLifeDayCount>
    <ClassCode>ZNOF</ClassCode>
    <ItemUsageTypeCode>00</ItemUsageTypeCode>
    <GlobalLifeDayCount>0</GlobalLifeDayCount>
    <ShelfLifeDayCountPercent>0</ShelfLifeDayCountPercent>
    <DayCountUOMCode>PCE</DayCountUOMCode>
    <MerchandisePlanningTypeCode>11</MerchandisePlanningTypeCode>
    <MainSupplierID>2041</MainSupplierID>
    <BaseUOMConsumerPackageGrossContent>0.000</BaseUOMConsumerPackageGrossContent>
    <BaseUOMConsumerPackageNetContent>34.000</BaseUOMConsumerPackageNetContent>
    <BaseUOMConsumerPackageContentsUOMCode>PCE</BaseUOMConsumerPackageContentsUOMCode>
    <BaseUOMConsumerPackageBasePriceContent>1</BaseUOMConsumerPackageBasePriceContent>

    <BaseUOMConsumerPackageBasePriceContentsUOMCode>PCE</BaseUOMConsumerPackageBasePriceContentsUOMCode>
    <DistributingWarehouseID>1</DistributingWarehouseID>
    <AssortmentID>T</AssortmentID>
    <ThirdPartyAssortmentID>1</ThirdPartyAssortmentID>
    <DistributionMethodID>1</DistributionMethodID>
    <TransportAssociationID>1</TransportAssociationID>
    <DefaultSellingUOMCode>PCE</DefaultSellingUOMCode>
    <ItemTextList>
      <ItemText>
        <TextClass>KTXT</TextClass>
        <Text>Winston Black BP 4XL 34er 10.00</Text>
        <TextNumber>1</TextNumber>
        <xx_custom_01>M</xx_custom_01>
        <xx_custom_10>18</xx_custom_10>
        <xx_custom_18>X</xx_custom_18>
      </ItemText>
    </ItemTextList>
    <SupplierItemList>
      <SupplierItem>
        <SupplierID>2041</SupplierID>
        <SupplierItemID>605984</SupplierItemID>
        <DefaultOrderUOMCode>K01</DefaultOrderUOMCode>
        <OrderUOMChoosableFlag>true</OrderUOMChoosableFlag>
        <PlannedDeliveryPeriod>8</PlannedDeliveryPeriod>
        <ShortDeliveryToleranceQuantity>0.0</ShortDeliveryToleranceQuantity>
        <OverDeliveryToleranceQuantity>0.0</OverDeliveryToleranceQuantity>
        <StatusCode>F</StatusCode>
        <MinimumOrderQuantity>0.000</MinimumOrderQuantity>
        <AdditionalOrderUOMList>
          <AdditionalOrderUOM>
            <UOMCode>K01</UOMCode>
          </AdditionalOrderUOM>
        </AdditionalOrderUOMList>
      </SupplierItem>
    </SupplierItemList>
    <UOMItemList>
      <UOMItem>
        <UOMCode>K01</UOMCode>
        <ConversionNumerator>8</ConversionNumerator>
        <ConversionDenominator>1</ConversionDenominator>
        <ConsumerPackageGrossWeight>0.000</ConsumerPackageGrossWeight>
        <ConsumerPackageNetWeight>0.000</ConsumerPackageNetWeight>
        <ConsumerPackageWeightUOMCode>KGM</ConsumerPackageWeightUOMCode>
        <ConsumerPackageVolume>0.000</ConsumerPackageVolume>
        <ConsumerPackageDepth>0.000</ConsumerPackageDepth>
        <ConsumerPackageWidth>0.000</ConsumerPackageWidth>
        <ConsumerPackageHeight>0.000</ConsumerPackageHeight>
        <HandlingTypeCode>X</HandlingTypeCode>
        <MainPOSItemID>4032800067583</MainPOSItemID>
        <StatusCode>F</StatusCode>
        <StatusEffectiveDate>1990-01-01</StatusEffectiveDate>
        <MaximumStockQuantity>0.000</MaximumStockQuantity>
        <MinimumStockQuantity>0</MinimumStockQuantity>
      </UOMItem>
    </UOMItemList>
  </Item>
</ItemList>

```

```

<ListingEffectiveDate>2021-04-11</ListingEffectiveDate>
<ListingExpirationDate>2099-12-31</ListingExpirationDate>
<MainMerchandiseHierarchyGroupID>1491</MainMerchandiseHierarchyGroupID>
<ItemSellingRule>
  <PriceEntryRequiredFlag>false</PriceEntryRequiredFlag>
  <ProhibitReturnFlag>false</ProhibitReturnFlag>
  <AllowFoodStampFlag>false</AllowFoodStampFlag>
  <WICFlag>false</WICFlag>
  <FSACode>0</FSACode>
  <DeliveryStockAssortmentFlag>false</DeliveryStockAssortmentFlag>
  <RetainPrepaymentFlag>false</RetainPrepaymentFlag>
  <TakeAwayAllowedFlag>false</TakeAwayAllowedFlag>
  <SerialNumberRequiredFlag>false</SerialNumberRequiredFlag>
  <TaxExemptProhibitedFlag>false</TaxExemptProhibitedFlag>
  <AuthorizedForSaleFlag>true</AuthorizedForSaleFlag>
  <DiscountFlag>false</DiscountFlag>
  <BonuspointsFlag>true</BonuspointsFlag>
  <QuantityInputTypeCode>00</QuantityInputTypeCode>
  <SellingEffectiveDate>2021-03-17</SellingEffectiveDate>
  <SellingExpirationDate>2099-12-31</SellingExpirationDate>
  <NegativePositionFlag>false</NegativePositionFlag>
  <PriceChangeTypeCode>02</PriceChangeTypeCode>
  <HidePriceFlag>true</HidePriceFlag>
  <ShowItemOnCustomerDisplayFlag>true</ShowItemOnCustomerDisplayFlag>
  <WeightEntryRequiredFlag>false</WeightEntryRequiredFlag>
  <PriceEntryOnScalesAllowedFlag>false</PriceEntryOnScalesAllowedFlag>
  <AllowLayawayFlag>true</AllowLayawayFlag>
  <LayawayDiscountFlag>true</LayawayDiscountFlag>
</ItemSellingRule>
<ItemMerchandiseManagementRule>
  <PriceEntryRequiredFlag>false</PriceEntryRequiredFlag>
  <QuantityInputTypeCode>00</QuantityInputTypeCode>
  <QuantityInputMethod>01</QuantityInputMethod>
  <NegativePositionFlag>false</NegativePositionFlag>
  <WeightEntryRequiredFlag>false</WeightEntryRequiredFlag>
  <DefaultLayoutClass>0</DefaultLayoutClass>
  <DefaultLabelCount>1</DefaultLabelCount>
  <ElectronicShelfLabelFlag>true</ElectronicShelfLabelFlag>
</ItemMerchandiseManagementRule>
<PriceList>
  <Price>
    <Price>80.00</Price>
    <EffectiveDate>2021-03-17T00:00:00</EffectiveDate>
    <ExpirationDate>2099-12-31T00:00:00</ExpirationDate>
    <PriceTypeCode>00</PriceTypeCode>
  </Price>
</PriceList>
<EANList>
  <EAN>
    <PosItemID>4032800067583</PosItemID>
    <PosIdentityTypeCode>HE</PosIdentityTypeCode>
    <PLCFlag>false</PLCFlag>
    <ScalesDomainID></ScalesDomainID>
    <PLCPart>4032800067583</PLCPart>
    <PLCPosItemID>4032800067583</PLCPosItemID>
  </EAN>
</EANList>
<SalesRestrictionList>
  <SalesRestriction>
    <SalesRestrictionTypeCode>CAGE</SalesRestrictionTypeCode>
    <SalesRestrictionValue>18</SalesRestrictionValue>
  </SalesRestriction>
</SalesRestrictionList>
<ItemDepositCollectionList></ItemDepositCollectionList>
<UOMItemTextList>
  <UOMItemText>
    <TextNumber>01</TextNumber>
    <TextClass>02</TextClass>
    <LanguageID>de_DE</LanguageID>
    <Text>Winston Black 4XL Stange</Text>
    <PictureFlag>false</PictureFlag>
  </UOMItemText>
  <UOMItemText>
    <TextNumber>900</TextNumber>
    <TextClass>08</TextClass>
    <LanguageID>de_DE</LanguageID>
    <Text>M</Text>
  </UOMItemText>

```

```

        <PictureFlag>false</PictureFlag>
      </UOMItemText>
    </UOMItemTextList>
    <ItemMHGList>
      <ItemMHG>
        <MerchandiseHierarchyGroupID>1491</MerchandiseHierarchyGroupID>
      </ItemMHG>
    </ItemMHGList>
    <ShelfAssignmentList>
      <ShelfAssignment>
        <ShelfID>11201330</ShelfID>
        <ShelfTierID>0000000314</ShelfTierID>
        <ShelfSpaceID>0000000314</ShelfSpaceID>
        <ShelfSpaceHorizontalPiecesCount>1</ShelfSpaceHorizontalPiecesCount>
        <ShelfSpaceVerticalPiecesCount>1</ShelfSpaceVerticalPiecesCount>
        <ShelfSpaceMaximumPiecesCount>1</ShelfSpaceMaximumPiecesCount>
        <ShelfSpaceOptimalPiecesCount>1</ShelfSpaceOptimalPiecesCount>
        <ShelfVersionID>0000</ShelfVersionID>
        <ShelfVersionEffectiveDate>2021-03-31</ShelfVersionEffectiveDate>
        <ShelfVersionExpirationDate>2099-12-31</ShelfVersionExpirationDate>
        <ShelfLabelType>0</ShelfLabelType>
        <ShelfSpaceSequence>0000000314</ShelfSpaceSequence>
        <xx_custom_01>X</xx_custom_01>
        <xx_custom_03>99991231</xx_custom_03>
        <xx_custom_04>20210331</xx_custom_04>
      </ShelfAssignment>
    </ShelfAssignmentList>
  </UOMItem>
</UOMItemList>
<ItemCollectionList></ItemCollectionList>
<ExternalModificationTimestamp>2021-06-15T20:18:36</ExternalModificationTimestamp>
<ExternalPackageID>0000000640966656</ExternalPackageID>
</Item>

```


Structure	XML Tag	Format	Example	Mandatory Field	Description	Validation and Error Code
<Item><UOMItemList><UOMItem>	<MainPOSItemID>	String	205000000102	x	Item Id	<ul style="list-style-type: none"> not set: 51011 item does already exist: 51029
<Item>	<ItemId>	String	12345	x	Material ID, Master ID (see master/variant handling)	
	<BaseUOMCode>	String	PCE	x	Base Quantity Unit	
<Item><UOMItemList><UOMItem><PriceList><Price>	<PriceTypeCode>	String	00	x	00 = <Price> is Regular Unit Price other: <Price> is Promotion Unit Price	<ul style="list-style-type: none"> invalid format or value < 0: 51016
	<Price>	Number in US format, no thousands separator, "." for comma	10.99	x	Depending on the value in <PriceTypeCode> Regular Unit Price or Promotion Unit Price is set.	<ul style="list-style-type: none"> invalid format or value < 0: 51016
	<PriceTypeCode>	String	00	x	00 = Regular Unit Price Any other → Promotion Unit Price	
	<EffectiveDate>	Timestamp yyyy-MM-dd'T'hh:mm:ss	2021-03-17T00:00:00	x	See details in chapter "Future Prices"	
	<ExpirationDate>	Timestamp yyyy-MM-dd'T'hh:mm:ss	2021-03-17T00:00:00	x	See details in chapter "Future Prices"	
<Item><UOMItemList><UOMItem>	<UOMCode>	String	PCE	x	Unit of Measure	
	<ConversionNumerator>	Integer	2		Item quantity set as: ConversionNumerator/ConversionDenominator	

	<ConversionDenominator>	Integer	1		Item quantity set as: ConversionNumerator/ConversionDenominator	
	<MainMerchandiseHierarchyGroupID>	String	1234		Assign the item to this category id	
<Item><UOMItemList><UOMItem><ItemSellingRule>	<SellingExpirationDate>	Date in format: yyyy-MM-dd	2022-01-01		Expiry date	If expiry date is infinity (9999-12-31) then the expiry date is not set.
<Item><UOMItemList><UOMItem><UOMItemTextList><UOMItemText>	<TextNumber>	String	100	x	See details "Mapping of Item Texts"	
	<TextClass>	String	02	x	See details "Mapping of Item Texts"	
	<LanguageID>	String ISO Language Code	de_DE	x	See details "Mapping of Item Texts"	
	<Text>	String		x	See details "Mapping of Item Texts"	

Configuration

General

Tag	Format	Example	Mandatory	Default	Description
languageCode	String	en		First language code is used.	Only the items and categories with the indicated Language Code are imported. You can configure more than one code, then separate the codes by comma: de,en If you do not provide the tag or the tag's value is empty, then the first language code that is provided is used.
defaultTaxonomy	String	Standard		Default	Name for the category taxonomy
importMode	INITIAL UPDATE	UPDATE		UPDATE	INITIAL = Deletes masterdata before the import starts: Items, item categories, categories, category hierarchies. UPDATE = Updates existing entities and adds new entities.

Text

Tag	Format	Example	Mandatory	Default	Description
languageId	String ISO Language Code	de_DE	x	empty	If not empty, import texts only for this language ID
textClasses	String, comma separated	01,02	x	empty	If not empty, import only texts having one of the named text classes.
excludedTextNumbers	String, comma separated	800,900	x	empty	If not empty, exclude text numbers named here

Future Prices

Tag	Format	Example	Mandatory	Default	Description
infinityDate	String dd.MM.yyyy	31.12.9999		31.12.9999	This date expresses infinity. If this date is set for a future price, then valid to is unbounded
nowOffset	Integer >= 0	0		1	Future begins from today + this offset. This is important for deciding to create a future price or not.
futurePriceImportEnabled	true, false	true		false	true = If future prices are found, import them false = If future prices are found, do NOT import them. However, import the first price that is found to be in valid from/to interval for the time of the import.

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

General

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

XML Reading

Please see details about the configuration here.

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <languageCode>de</languageCode>
  <importMode>UPDATE</importMode>
  <infinityDate>31.12.2099</infinityDate>
  <nowOffset>0</nowOffset>
  <futurePriceImportEnabled>false</futurePriceImportEnabled>
  <xmlReader>
    <charset>UTF-8</charset>
  </xmlReader>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>Item</mappingId>
      <regExpression>Items.*</regExpression>
      <type>Item</type>
      <defaultName>Items.xml</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <!-- httpInboundAdapter>
    <waitOnDownloadFailedInSeconds>5</waitOnDownloadFailedInSeconds>
    <retryCounter>0</retryCounter>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <httpResources>
      <httpResource>
        <resourceId>TransactionLog</resourceId>
        <resourceUrl>http://test.de</resourceUrl>
        <query></query>
        <batchSize>0</batchSize>
        <type>TransactionLog</type>
        <authentication>
          <user>test</user>
          <encryptedPassword>test</encryptedPassword>
        </authentication>
      </httpResource>
    </httpResources>
  </httpInboundAdapter-->
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

2.5.11 Exchange Rate Import

2.5.11.1 Exchange Rate Import - CSV

Summary

Exchange rates describe the conversion between different currencies.

The exchange rates of the currencies are imported via a **CSV file**.

The exchange rates can be used in both directions, in this case with reverse exchange rate.

Communication

The following options are supported for the communication:

Trigger	Processing	Description
GET		<ul style="list-style-type: none">Ad hoc reading of one or more files from the file system.Ad hoc querying of data from an endpoint via HTTP GET or POST.
POST		<ol style="list-style-type: none">A file is transmitted.One or more files in the ZIZ archive are transmitted.
Scheduler		<ul style="list-style-type: none">Time-controlled reading of one or more files from the file system.Time-controlled querying of a file from an endpoint via HTTP GET or POST.

Data

Example

```
baseCurrency|quoteCurrency|quote|effectiveDate  
EUR|USD|1.14|2018-01-05
```

Mapping and Validation

CSV field	Format	Example	Mandatory Field	Mapping	Validation and Error Code
baseCurrency	ISO 4217*	EUR	x		<ul style="list-style-type: none">not set: 51011invalid value: 51042
quoteCurrency	ISO 4217*	USD	x		<ul style="list-style-type: none">not set: 51011invalid value: 51042
quote	Double	1.14	x		<ul style="list-style-type: none">not set: 51011invalid value: 51030
effectiveDate	YYYY-MM-dd	2018-01-05	x		<ul style="list-style-type: none">not set: 51011invalid value: 51015

*) ISO 4217 for currency code: https://de.wikipedia.org/wiki/ISO_4217

Validity date

The exchange rates have an activation date. This date must be the same for all IDoc entries. Otherwise, all entries will be discarded.

Configuration

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

CSV Reading

Please see details about the configuration here.

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>Competitor</mappingId>
      <regExpression>competitors.*csv</regExpression>
      <type>Competitor</type>
      <defaultName>competitors.csv</defaultName>
    </inboundDataMapping>
    <inboundDataMapping>
      <mappingId>CompetitorPrice</mappingId>
      <regExpression>competitorprice.*csv</regExpression>
      <type>CompetitorPrice</type>
      <defaultName>competitorprice.csv</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <csvReader>
    <charset>UTF-8</charset>
    <readHeader>true</readHeader>
    <escapeMode>BACKSLASH</escapeMode>
    <columnSeparator>|</columnSeparator>
  </csvReader>
  <fileInboundAdapter>
    <importDir>import</importDir>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <copyOption>NONE</copyOption>
    <isLockFileEnabled>false</isLockFileEnabled>
    <lockFileName>lock</lockFileName>
    <triggerOption>NO_TRIGGER</triggerOption>
    <triggerFileName>trigger</triggerFileName>
    <waitOnFileValidationFailedInSeconds>5</waitOnFileValidationFailedInSeconds>
    <fileValidationRetryCount>0</fileValidationRetryCount>
  </fileInboundAdapter>
</import-config>
```

Code Block 24 Sample configuration

2.5.11.2 Exchange Rate Import - SAP Enterprise Service

Summary

Exchange rates describe the conversion between different currencies.

The exchange rates for currencies are imported via the SAP Enterprise Service ExchangeRateByExchangeRateQuery_sync.

The exchange rates can be used in both directions, in this case with reverse exchange rate.

Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET		<ul style="list-style-type: none">• Ad hoc reading of one or more files from the file system.• Ad hoc sending of a request to an SAP system via POST.
POST		<ol style="list-style-type: none">1. A file is transmitted.2. One or more files in the ZIZ archive are transmitted.
Scheduler		<ul style="list-style-type: none">• Time-controlled reading of one or more files from the file system.• Time-controlled sending of a request to an SAP system via POST.

Query

A request is made by POST request. This request is configurable.

```
<?xml version="1.0" encoding="UTF-8"?>
<ns1:ExchangeRateByExchangeRateQuery_sync xmlns:ns1="http://sap.com/xi/APPL/SE/Global">
  <MessageHeader>
    <CreationDateTime>2015-10-21 08:44:26</CreationDateTime>
  </MessageHeader>
  <ExchangeRateSelectionByExchangeRate>
    <ExchangeRateTypeCode>*</ExchangeRateTypeCode>
    <UnitCurrency>*</UnitCurrency>
    <QuotedCurrency>*</QuotedCurrency>
    <ValidFromDate>2018-10-01T08:44:25Z</ValidFromDate>
  </ExchangeRateSelectionByExchangeRate>
</ns1:ExchangeRateByExchangeRateQuery_sync>
```

Data

An XML response to the query is sent. This may also be transmitted as a file:

```
<nm:ExchangeRateByExchangeRateResponse_sync xmlns:nm="http://sap.com/xi/APPL/SE/Global"
xmlns:prx="urn:sap.com:proxy:Q47:/1SAI/TAS9551CB18EB41D2E39F04:731" xmlns:soap-
env="http://schemas.xmlsoap.org/soap/envelope/">
  <MessageHeader>
    <CreationDateTime>2015-10-21T10:44:11Z</CreationDateTime>
  </MessageHeader>
  <ExchangeRate>
    <UnitCurrency>EUR</UnitCurrency>
    <QuotedCurrency>USD</QuotedCurrency>
    <Rate>1.0</Rate>
    <QuotationDateTime>2015-10-21T00:00:00Z</QuotationDateTime>
  </ExchangeRate>
  <ExchangeRateTypeCode>EURX</ExchangeRateTypeCode>
  <Log/>
</nm:ExchangeRateByExchangeRateResponse_sync>
```

Mapping and Validation

XML field	Format	Example	Mandatory Field	Mapping	Validation and Error Code
UnitCurrency	ISO 4217*	EUR	x		<ul style="list-style-type: none">• nicht vorhanden: 51011• ungültiger Wert: 51042
QuotedCurrency	ISO 4217*	USD	x		<ul style="list-style-type: none">• nicht vorhanden: 51011• ungültiger Wert: 51042
Rate	Double	1.14	x		<ul style="list-style-type: none">• nicht vorhanden: 51011• ungültiger Wert: 51030
QuotationDateTime	YYYY-MM-ddTHH:mm:ssZ	2018-01-05T00:00:00Z	x		<ul style="list-style-type: none">• nicht vorhanden: 51011• ungültiger Wert: 51015

*) ISO 4217 for currency code: https://de.wikipedia.org/wiki/ISO_4217

Configuration

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

XML Reading

Please see details about the configuration here.

Example

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<import-config>
  <exchangeRateTypeCode>EURX</exchangeRateTypeCode>
  <inboundDataMappings>
    <inboundDataMapping>
      <mappingId>ExchangeRate</mappingId>
      <regExpression>exchangerates.*xml</regExpression>
      <type>ExchangeRate</type>
      <defaultName>exchangerates.xml</defaultName>
    </inboundDataMapping>
  </inboundDataMappings>
  <xmlReader>
    <charset>UTF-8</charset>
  </xmlReader>
  <httpInboundAdapter>
    <waitOnDownloadFailedInSeconds>5</waitOnDownloadFailedInSeconds>
    <retryCounter>0</retryCounter>
    <isArchiveToTaskEnabled>true</isArchiveToTaskEnabled>
    <httpResources>
      <httpResource>
        <resourceId>ExchangeRate</resourceId>
        <resourceUrl><![CDATA[http://sap-pq-pi.gk.gk-
software.com:50000/XISOAPAdapter/MessageServlet?senderService=BS_PRUDSYS_PRICING3&interface=Exc
hangeRateByExchangeRateQueryResponse_In&interfaceNamespace=http://sap.com/xi/APPL/SE/Global]]><
/resourceUrl>
        <httpMethod>POST</httpMethod>
        <contentType>text/xml</contentType>
        <body><![CDATA[<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:glob="http://sap.com/xi/APPL/SE/Global">
  <soapenv:Header/>
  <soapenv:Body>
    <glob:ExchangeRateByExchangeRateQuery_sync>
      <MessageHeader>
        <CreationDateTime>2015-10-21 08:44:26</CreationDateTime>
      </MessageHeader>
      <ExchangeRateSelectionByExchangeRate>
        <ExchangeRateTypeCode>EURX</ExchangeRateTypeCode>
        <UnitCurrency>*</UnitCurrency>
        <QuotedCurrency>*</QuotedCurrency>
      </ExchangeRateSelectionByExchangeRate>
      </glob:ExchangeRateByExchangeRateQuery_sync>
    </soapenv:Body>
  </soapenv:Envelope>]]></body>
        <type>ExchangeRate</type>
        <socketTimeoutInSeconds>5</socketTimeoutInSeconds>
        <connectionTimeoutInSeconds>5</connectionTimeoutInSeconds>
        <!-- cloudConnector>
          <destination></destination>
        </cloudConnector-->
        <authentication>
          <user>PRUDSYS</user>
          <encryptedPassword>WjXESa1amzSKcz</encryptedPassword>
        </authentication>
      </httpResource>
    </httpResources>
  </httpInboundAdapter>
</import-config>
```

Code Block 25 Sample configuration

2.5.12 Business Unit Import

2.5.12.1 Summary

This import allows you to import business units and business unit hierarchy in XML format.

Supported use cases:

- Create business units

- Create business units hierarchy
- Update data
- Create business units with repetitions, e.g. 100 stores in parent business unit "global"
- Move business units from one parent to another
- Derive settings from parent business unit


2.5.12.2 Communication

Options


The following options are supported for the communication:

Trigger	Processing	Description
GET	Previous deletion of the current data	Ad hoc reading of the files from the file system or by an HTTP endpoint
POST	Previous deletion of the current data	The files in the ZIP archive are transferred
Scheduler	Previous deletion of the current data	Time-controlled reading of the files from the file system or from an HTTP endpoint

2.5.12.3 Modes of Import

Mode	Description
INITIAL	Deletes all business units before starting to import new ones.  Exception: The business unit that runs the import is NOT deleted.
UPDATE	Updates business units. In this mode you can: <ul style="list-style-type: none"> • Add new business units • Move BUs • Remove BUs • Delete subtrees of the BU hierarchy

 The default mode is UPDATE.

 You need to create a business unit to configure the business unit import. This is needed as bootstrap configuration. You can create a technical BU just for import or you create the root BU and then use the update or initial mode to setup the rest. Please mind that in initial mode this BU is not deleted.

High Level Data Structure


```

<BusinessUnitImport Version="1.0.0">
  <BusinessUnits>
    <BusinessUnit>
      <ControlHeader>
        ...
      </ControlHeader>
      <Identification>
        <Name>Headquarters</Name>
        <Id>hq</Id>
      </Identification>
    </BusinessUnit>
    <BusinessUnit>
      <Identification>
        <Name>Store 123</Name>
        <Id>store123</Id>
        <Parent>hq</Parent>
      </Identification>
      <SafetyNetRules>
        ...
      </SafetyNetRules>
      <Data>
        <CurrencyCode>USD</CurrencyCode>
        ...
      </Data>
      <Parameters>
        ...
      </Parameters>
    </BusinessUnit>
  </BusinessUnits>
</BusinessUnitImport>

```

Tag	Mandatory	Description
BusinessUnitImport	x	Root element. Also states the version of the imported data.
BusinessUnits	x	Root element for the business unit list
BusinessUnit	x	Data for business unit
ControlHeader		You can control: <ul style="list-style-type: none"> Delete BUs Generate several BUs using an iteration
Identification	x	Identification of the BU
SafetyNetRules		Assign Safety Net rules
Data	x	Data including global rounding rules
Parameters		Parameter of a Business Unit

2.5.12.4 Section: ControlHeader

Iterations

You can create a bunch of BUs or update a bunch of them at once.

```

<BusinessUnit>
  <ControlHeader>
    <Iteration>
      <From>1</From>
      <To>100</To>
    </Iteration>
  </ControlHeader>
  <Identification>
    <Name>Headquarters</Name>
    <Id>store${Iteration}</Id>
    <Parent>hq</Parent>
  </Identification>
</BusinessUnit>

```

XML Tag	Format	Example	Mandatory	Default	Description	Validation and Error Codes
From	Integer or String Integer: 0 - MaxInt String: a-z, A-Z	1 a	x		Start of the range	
To	Integer or String Integer: 0 - MaxInt String: a-z, A-Z	5 c	x		End of the range	

For each iteration you get this parameter that you can use for all tags in the Identification section: Name, Id, Parent.

The parameter is replaced with the current iteration value:

- \${Iteration}

Example for range 1-10 and first iteration for the Id tag:

```
<Name>Store ${Iteration}</Name>
<Id>store${Iteration}</Id>
```

is replaced with:

```
<Name>Store 1</Name>
<Id>store1</Id>
```

Update Mode: Delete

You can delete business units in several ways:

- Delete a business unit. If there are child business units, all of them are deleted as well
- Delete the children of a business unit recursively

In the protocol you can find an event naming all deleted business units.

For deleting a business unit, it's enough to:

- Use the ControlHeader with Delete mode
- Add the ID in the identification section

Delete does also support iterations.

```
<BusinessUnit>
  <ControlHeader>
    <DeleteMode>DELETE</DeleteMode>
  </ControlHeader>
  <Identification>
    <Id>hq</Id>
  </Identification>
</BusinessUnit>
```

XML Tag	Format	Example	Mandatory	Default	DescriptionBeschreibung	Validation and Error Code
DeleteMode	DELETE DELETE_CHILDREN	DELETE			DELETE: Delete the current BU and all child BUs DELETE_CHILDREN: Delete only the child BUs	

Section: Identification

```
<BusinessUnit>
  <Identification>
    <Name>Headquarters</Name>
    <Id>hq</Id>
  </Identification>
  ...
</BusinessUnit>
```

XML Tag	Format	Example	Mandatory	Default	Description	Validation and Error Code
Name	String	Headquarters	x		Name of the business unit	
Id	String	hq	x		ID of the business unit. Can not be updated later on.	

Parent/Child Relationship

```

<BusinessUnit>
  <Identification>
    <Name>Headquarters</Name>
    <Id>hq</Id>
  </Identification>
  ...
</BusinessUnit>
<BusinessUnit>
  <Identification>
    <Name>Store 101</Name>
    <Id>store101</Id>
    <Parent>hq</Parent>
  </Identification>
  ...
</BusinessUnit>

```

XML Tag	Format	Example	Mandatory	Default	Description	Validation and Error Code
Parent	String	region			Link to a parent business unit. This is optional.	

Changing the Parent

In the update mode you are able to change the parent. So you could move a store to another region.

```

<BusinessUnit>
  <Identification>
    <Name>Region A</Name>
    <Id>regionA</Id>
  </Identification>
  ...
</BusinessUnit>
<BusinessUnit>
  <Identification>
    <Name>Region B</Name>
    <Id>regionB</Id>
  </Identification>
  ...
</BusinessUnit><BusinessUnit>
  <Identification>
    <Name>Store 101</Name>
    <Id>store101</Id>
    <Parent>regionB</Parent>
  </Identification>
  ...
</BusinessUnit>

```

If a business unit A is moved to another parent B and you are using data inheritance, then the data inheritance might change the data of A and all of its children (if they also use inheritance).

2.5.12.5 Section: Data

Data Inheritance

You can inherit parts or all data from a parent business unit. If the BU does not exist, an error is thrown.

To inherit all data from the parent:

```
<Data InheritFromParent="true"/>
```

Just inherit a part of the data, set currency explicitly and use the defaults for the rest:

```
<Data InheritFromParent="true">
  <CurrencyCode>EUR</CurrencyCode>
</Data>
```

Data Fields

```
<Data>
  <CurrencyCode>USD</CurrencyCode>
  <TimeZone>GMT+00:00</TimeZone>
  <ComputationsEnabled>true</ComputationsEnabled>
  <IncludeTransactionsOfOtherBusinessUnit>hq</IncludeTransactionsOfOtherBusinessUnit>
  <PresetReviewsForComputationEnabled>false</PresetReviewsForComputationEnabled>
  <SegmentStatisticsSchedulerTimeStamp>00:00</SegmentStatisticsSchedulerTimeStamp>
  <TurnoverStatisticsSchedulerTimeStamp>00:00</TurnoverStatisticsSchedulerTimeStamp>
  <DisplayNetTurnoverEnabled>false</DisplayNetTurnoverEnabled>
  <PriceNearBorderThreshold>15</PriceNearBorderThreshold>
  <StrategyStatisticField1>regularPrice</StrategyStatisticField1>
  <StrategyStatisticField2>recommendedPrice</StrategyStatisticField2>
  <CleanupSegmentStatisticsOlderThanDays>30</CleanupSegmentStatisticsOlderThanDays>
  <CleanupSimulationsOlderThanDays>5</CleanupSimulationsOlderThanDays>
  <CleanupStrategyStatisticsOlderThanDays>30</CleanupSegmentStatisticsOlderThanDays>
  <CleanupTransactionsOlderThanDays>30</CleanupTransactionsOlderThanDays>
  <CleanupMasterdataOlderThanRevisions>30</CleanupMasterdataOlderThanRevisions>
  <CleanupComputationsOlderThanDays>7</CleanupComputationsOlderThanDays>
  <CleanupImportsOlderThanDays>7</CleanupImportsOlderThanDays>
  <CleanupExportsOlderThanDays>7</CleanupExportsOlderThanDays>
  <CleanupSchedulerTimeStamp>00:00</CleanupSchedulerTimeStamp>
  <CleanupItems>44</CleanupItems>
  <CleanupItemsDeleteTranslogs>false</CleanupItemsDeleteTranslogs>
  <RoundingRules>...</RoundingRules>
</Data>
```

XML Tag	Format	Example	Mandatory	Default	Description	Validation and Error Code
CurrencyCode	String, see ISO 4217	USD	x		Master Currency Code of the business unit	<ul style="list-style-type: none"> Error Unknown Code
TimeZone	Timezone Format	GMT+01:00	x		Master timezone. Used for importing transactions, exporting prices.	<ul style="list-style-type: none"> Error Unable to parse timezone
ComputationsEnabled	true, false	true		true	Should we compute prices for this business unit. This can be disabled for pure organizational business units.	
IncludeTransactionsOfOtherBusinessUnit	String: id of another business unit	hq		no includes	You can reference transactions from other business unit and her children	<ul style="list-style-type: none"> Warning if referenced business unit does not exist
PresetReviewsForComputationEnabled	true, false	true		false	A preset for computations: If created, should review flag set to enabled?	
SegmentStatisticsSchedulerTimeStamp	Timestamp in 24 hours format	00:00		00:00	Timestamp when the segment statistic should be calculated	
TurnoverStatisticsSchedulerTimeStamp	Timestamp in 24 hours format	00:00		00:00	Timestamp when the turnover statistic should be calculated	
DisplayNetTurnoverEnabled	true, false			false		
PriceNearBorderThreshold	1..50			disabled	Value in percent: when should a price be treated as near a border or not.	
StrategyStatisticField1	Price fields of the item				See list of fields*	<ul style="list-style-type: none"> Error if field name does not exist or is not allowed
StrategyStatisticField2	Price fields of the item				See list of fields*	<ul style="list-style-type: none"> Error if field name does not exist or is not allowed
CleanupSegmentStatisticsOlderThanDays	1...max integer	25		30	Delete segment statistics that are older than x days.	
CleanupSimulationsOlderThanDays	1...max integer			5	Delete simulations older than x days.	
CleanupStrategyStatisticsOlderThanDays	1...max integer			30	Delete strategy statistics older than x days.	
CleanupTransactionsOlderThanDays	1...max integer			30	Delete transactions older than x days.	
CleanupMasterdataOlderThanRevisions	1...max integer			30	Delete master data revisions older than x revisions. Each import writes a new revision.	
CleanupComputationsOlderThanDays	1...max integer			7	Delete computation results older than x days.	
CleanupImportsOlderThanDays	1...max integer			7	Delete imports older than x days. Only the protocol and the (optionally) saved import data is deleted.	
CleanupExportsOlderThanDays	1...max integer			7	Delete exports older than x days. Only the (optionally) saved exported data and some metadata is deleted.	
CleanupSchedulerTimeStamp	Timestamp in 24 hours format	00:00		00:00	When to run the cleanup task.	

CleanupItems	1..max integer	30			<p>If you don't want to delete the history of the items, but the entire item including its history if it is too old. What means too old?</p> <p>If you first import an item that did not exist in the database before a created date is maintained for the item (this is true for both initial and update mode setting). If you update your item later on, the created item does not change. To delete the item that is older than 365 days the housekeeping process checks if there is an item with a created date older than 365 days from now. If it is older, the item including all versions are deleted.</p> <p>Example:</p> <ul style="list-style-type: none"> Item created date is 20.01.2021 Keep items that are newer than 4 days Current time is 25.01.2021 15:00 and the scheduler starts <p>First the current time is moved to 25.01.2021 00:00. Then 4 days are subtracted: 25.01.2021 - 4 days = 21.01.2021.</p> <p>Now the created date of the items is compared to it: 20.01.2021 < 21.01.2021? Yes, so delete the item and the version history.</p>	
CleanupItemsDeleteTranslogs	true, false	true		false	<p>If you turn that on, also transactions that reference this item are deleted.</p> <p>This setting works only if you set a value for CleanupItems (see above)</p>	

* List of fields for statistics

- regularUnitPrice
- minUnitPrice
- maxUnitPrice
- promotionUnitPrice
- minPricingLimit
- maxPricingLimit
- purchaseUnitPrice
- recommendedRetailPrice
- And all extension fields "x_NNN" of type money

You can add your own data fields via the SDK extensions. This is the preferred way to add new data as you can then see data field names in filters and in the master data explorer.

Rounding

```
<Data>
...
<RoundingRules IsMasterConfiguration="false">
  <RoundingRule>
    <Name>My_rule</Name>
    <RoundingPreset>NONE</RoundingPreset>
    <PriceLimitFrom>13.00</PriceLimitFrom>
    <PriceLimitTo>29.00</PriceLimitTo>
    <RoundingMode>ROUNDUP</RoundingMode>
    <RoundingDefinitions>
      <RoundingDefinition>
        <DecimalPlacesBeforeComma Count="6"/>
        <DecimalPlacesAfterComma Count="2">
          <DecimalPlace Index="1">1,5</DecimalPlace>
          <DecimalPlace Index="2">9</DecimalPlace>
        </DecimalPlacesAfterComma>
      </RoundingDefinition>
    </RoundingDefinitions>
  </RoundingRule>
</RoundingRules>
</Data>
```

RoundingRules

RoundingRules	Format	Example	Mandatory	Default	Description	Validation and error code
isMasterConfiguration	true, false	true		false	true = Die Konfiguration wird an die Kind-BUSs vererbt false = Die Konfiguration wird NICHT vererbt	

RoundingRule

You can create one or more rounding rules.

RoundingRule	Format	Example	Mandatory	Default	Description	Validation and error code
Name	String	My Rule	x		Name of the rounding rule. We do not check if the name already exists.	Error if not set or empty
RoundingPreset	NONE CURRENCYDECIMALS CASHROUNDING	CASHROUNDING	x		NONE = Define a custom rounding rule. CURRENCYDECIMALS = Rounding for digits after comma defined by the configured currency CASHROUNDING = In addition to CURRENCYDECIMALS digits after the comma uses the currency defined setting (e.g. rounding of rappen in Switzerland) In case of using a preset, then you can not define a custom rounding rule.	
PriceLimitFrom	Number no thousands separator, "." for comma	12.99		No limit	Define the lower boundary of a computed price that triggers this rule.	
PriceLimitTo	Number no thousands separator, "." for comma	15.99		No limit	Define the upper boundary of a computed price that triggers this rule.	
RoundingMode	ROUNDUP ROUNDDOWN ROUNDDISTANCE	ROUNDUP		ROUNDUP	ROUNDUP = Round up ROUNDDOWN = Round down ROUNDDISTANCE = calculates distance to next smaller and next greater price and then rounds up from 50% of the interval. Works similar like the mathematical rounding, however works with a flexible value range.	

In both initial and update mode you need to name all rounding rules. So if you initially created a business unit with two rounding rules and want to update the second rule, you need to put there rule 1 and the changed rule 2 to the XML file. If you just put rule 2 to the XML file, then rule 1 will be deleted.

RoundingDefinition

You can define rounding definitions only in case of RoundingPreset=NONE. Otherwise they will be discarded.

You can configure rounding for 8 decimal digits of a price.

You can configure the digits before and after comma with the "Count" attribute:

```
<RoundingDefinition>
  <DecimalPlacesBeforeComma Count="6"/>
  <DecimalPlacesAfterComma Count="2">...
</DecimalPlacesAfterComma>
</RoundingDefinition>
```


RoundingDefinition	Format	Example	Mandatory	Default	Description	Validation and Error Code
DecimalPlacesBeforeComma Count=".."	Count: Integer 1..7	2		If you just set DecimalPlacesAfterComma, then 7 - DecimalPlacesAfterComma "Count"	Count of digits before comma	<ul style="list-style-type: none"> Only 1-7 is allowed Error if Before + After > 7
DecimalPlacesAfterComma Count=".."	Count: Integer 1..7	2		If you just set DecimalPlacesBeforeComma then 7 - DecimalPlacesBeforeComma "Count"	Count of digits after comma	<ul style="list-style-type: none"> Only 1-7 is allowed Error if Before + After > 7
<DecimalPlacesBeforeComma> <DecimalPlace Index="..">	Index: <ul style="list-style-type: none"> Integer 1..8 More values separated by comma: 1,2,3 More values using a range: 1-4 Decimal Place: <ul style="list-style-type: none"> Integer 0..9 More values separated by comma 	<DecimalPlace Index="1"> 5</DecimalPlace> <DecimalPlace Index="1"> 1,5</DecimalPlace> <DecimalPlace Index="1-2"> 1,5</DecimalPlace>			Round to a value for a digit before the comma. You can define more values for a digit: <ul style="list-style-type: none"> "1,2,3" "1-3" Index means place before the comma , starting with 1 left to the comma .	

<code><DecimalPlacesAfterComma></code> <code><DecimalPlace Index=".."></code>	Index: <ul style="list-style-type: none"> Integer 1..8 More values separated by comma: 1,2,3 More values using a range: 1-4 Decimal Place: <ul style="list-style-type: none"> Integer 0..9 More values separated by comma 	<code><DecimalPlace Index="1"></code> <code>5</DecimalPlace></code> <code><DecimalPlace Index="1"></code> <code>1,5</DecimalPlace></code> <code>></code> <code><DecimalPlace Index="1-2"></code> <code>1,5</DecimalPlace></code> <code>></code>			Round to a value for a digit after the comma. You can define more values for a digit: <ul style="list-style-type: none"> "1,2,3" "1-3" Index means place after the comma, starting with 1 after the comma.	
--	--	--	--	--	---	--

2.5.12.6 Business Unit: Parameters

It is possible to define business unit parameters. If someone asks for a parameter and its value for a given business unit, then the besides the current node in the hierarchy also a backtracking upwards is implemented. Explicit data inheritance is not needed here.

```

<Parameters>
  <Parameter Name="DistributionChain">RS10</Parameter>
</Parameters>

```

XML Tag	Format	Mandatory	Description	Validation
Parameter, Attribute: Name	String		Name and value of the parameter. Value can be empty, parameter name not.	<ul style="list-style-type: none"> Name is mandatory

Update handling: If you want to add a new parameter, but want to keep the existing ones, then you need to name the existing parameters as well. If you define just one parameter, all others will be discarded. If your list is empty, all parameters are discarded. If there are no Parameters tags at all, no changes to parameters are applied.

Delete all parameters:

```

<Parameters>
</Parameters>

```

Assigning Safety Net Rules

For each business rules you can assign safety net rules:

```

<SafetyNetRules>
  <SafetyNetRule>
    <Id>CalculatedPrice</Id>
    <Parameters>
      <Parameter Name="operator">GREATER</Parameter>
      <Parameter Name="value">0.0</Parameter>
    </Parameters>
  </SafetyNetRule>
</SafetyNetRules>

```

XML Tag	Format	Example	Mandatory	Default	Description	Validation and error code
Id	String	CalculatedPrice	x		Business ID of the safety net rule	• Unknown ID
Parameter, Name	String		x		Key, Value for a parameter of a safety net rule	

Standardmäßig werden Safety Net Regeln, die als default gekennzeichnet sind, angelegt. Man kann das Anlegen der Regeln verhindern:

```

<SafetyNetRules CreateDefaultSafetyNetRules="false">

```

Es ist auch möglich, die Regeln vom Vater-Knoten zu übernehmen:

```

<SafetyNetRules InheritFromParent="true" />

```

Standard Safety Nets

Besides the standard safety nets you can also use custom safety nets created by your custom extension.

Standard Safety Net: Minimal or Maximal Price

Tag	Value	Format	Example	Mandatory	Description	Validation and Error Code
Id	CalculatedPrice	Fixed String	CalculatedPrice	x	ID of the safety net rule	• Unknown ID
Parameter Name	operator	LOWER GREATER GREATER_EQUALS LOWER_EQUALS	GREATER	x	Operator for comparing the computed price with.	• Unknown operator
Parameter Name	value	Number no thousands separator, "." for comma	0.0	x	Value for the operator	• Not a number

Example:

```

<SafetyNetRules>
  <SafetyNetRule>
    <Id>CalculatedPrice</Id>
    <Parameters>
      <Parameter Name="operator">GREATER</Parameter>
      <Parameter Name="value">0.0</Parameter>
    </Parameters>
  </SafetyNetRule>
</SafetyNetRules>

```

Standard Safety Net: Checking Price Limits

Tag	Value	Format	Example	Mandatory	Description	Validation and Error Code
Id	PricingLimits	Fixed String	PricingLimits	x	ID of the safety net rule	• Unknown ID
Parameter Name	validateMaxPrice	true false	true	x	Compare the computed price with the item's maxUnitPrice Limit. If the price is lower or equals, then the item passes the safety net.	• Unknown value
Parameter Name	validateMinPrice	true false	false	x	Compare the computed price with the item's minUnitPrice Limit. If the price is greater or equals, then the item passes the safety net.	• Unknown value

Example:

```
<SafetyNetRules>
  <SafetyNetRule>
    <Id>PricingLimits</Id>
    <Parameters>
      <Parameter Name="validateMaxPrice">false</Parameter>
      <Parameter Name="validateMinPrice">true</Parameter>
    </Parameters>
  </SafetyNetRule>
</SafetyNetRules>
```

2.5.12.7 Example

```
<BusinessUnitImport Version="1.0.0">
  <BusinessUnits>
    <BusinessUnit>
      <Identification>
        <Name>Headquarters</Name>
        <Id>hq</Id>
      </Identification>
    </BusinessUnit>
    <BusinessUnit>
      <Identification>
        <Name>Store</Name>
        <Id>store1</Id>
        <Parent>hq</Parent>
      </Identification>
      <Data>
        <CurrencyCode>USD</CurrencyCode>
        <TimeZone>GMT+00:00</TimeZone>
        <RoundingEnabled cashRounding="false">true</RoundingEnabled>
        <ComputationsEnabled>true</ComputationsEnabled>
        <IncludeTransactionsOfOtherBusinessUnit>123</IncludeTransactionsOfOtherBusinessUnit>
        <PresetReviewsForComputationEnabled>false</PresetReviewsForComputationEnabled>
        <SegmentStatisticsSchedulerTimeStamp>00:00</SegmentStatisticsSchedulerTimeStamp>
        <TurnoverStatisticsSchedulerTimeStamp>00:00</TurnoverStatisticsSchedulerTimeStamp>
        <DisplayNetTurnoverEnabled>false</DisplayNetTurnoverEnabled>
        <PriceNearBorderThreshold>15</PriceNearBorderThreshold>
        <StrategyStatisticField1>regularPrice</StrategyStatisticField1>
        <StrategyStatisticField2>recommendedPrice</StrategyStatisticField2>
        <CleanupSegmentStatisticsOlderThanDays>30</CleanupSegmentStatisticsOlderThanDays>
        <CleanupSimulationsOlderThanDays>5</CleanupSimulationsOlderThanDays>
        <CleanupStrategyStatisticsOlderThanDays>30</CleanupStrategyStatisticsOlderThanDays>
        <CleanupTransactionsOlderThanDays>30</CleanupTransactionsOlderThanDays>
        <CleanupMasterdataOlderThanDays>30</CleanupMasterdataOlderThanDays>
        <CleanupComputationsOlderThanDays>7</CleanupComputationsOlderThanDays>
        <CleanupImportsOlderThanDays>7</CleanupImportsOlderThanDays>
        <CleanupExportsOlderThanDays>7</CleanupExportsOlderThanDays>
        <CleanupSchedulerTimeStamp>00:00</CleanupSchedulerTimeStamp>
        <CleanupItems>40</CleanupItems>
        <CleanupItemsDeleteTranslogs>true</CleanupItemsDeleteTranslogs>
      </Data>
    </BusinessUnit>
  </BusinessUnits>
</BusinessUnitImport>
```

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File SystemGeneral

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

2.5.13 Product Link Import

2.5.13.1 Summary

This document describes the import of product relations in XML format.

You can import:

- Family pricing groups
- Family pricing product analyzers
- Quantity pricing analyzers
- Product link relations

2.5.13.2 Communication

Options

The following options are supported for the communication:

Trigger	Processing	Description
GET	Previous deletion of the current data	Ad hoc reading of the files from the file system or by an HTTP endpoint
POST	Previous deletion of the current data	The files in the ZIP archive are transferred
Scheduler	Previous deletion of the current data	Time-controlled reading of the files from the file system or from an HTTP endpoint

2.5.13.3 Top Level Structure

```

<ProductRelationImport Version="1.0.0">

  <FamilyRelations>

    <ManualFamily>
    </ManualFamily>

    <AttributeFamily>
    </AttributeFamily>

    <CustomFamily Type="">
    </CustomFamily>

  </FamilyRelations>

  <QuantityRelations>

    <QuantityRelation>
    </QuantityRelation>

  </QuantityRelations>

  <ItemLinkRelations>
    <ItemLinkRelationGroup>
    </ItemLinkRelationGroup>
  </ItemLinkRelations>

</ProductRelationImport>

```

Tag	Mandatory	Description
ProductRelationImport	x	Root element for the importer. Includes the version number. If the version is incompatible then the importer stops emitting an error in the import protocol.
FamilyRelations		Same price relations
QuantityRelations		Quantity relations
ItemLinkRelations		Link relations

2.5.13.4 Mapping and Validation

Mapping: Family Relations

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<ProductRelationImport Version="1.0.0">

  <FamilyRelations>
    <ManualFamily>
      <ManualGroups>
        <ManualGroup>
          <ID>ManualGroup_1</ID>
          <ReferenceItem>MR318412</ReferenceItem>
          <Item>MR318412</Item>
          <Item>MR31841220</Item>
          <Item>MR31841230</Item>
          <Item>MR31841240</Item>
        </ManualGroup>
      </ManualGroups>
    </ManualFamily>

    <AttributeFamily>
      <AttributeAnalyzers>
        <AttributeAnalyzer>
          <ID>AttributeAnalyzer_12345</ID>
          <Name>Test Attribute Update</Name>
          <Description>Test Description</Description>
          <Mergeable>true</Mergeable>

          <Filter>{"condition":{"operands":[{"operands":[{"operands":[],"active":true,"attributes":"","value":["i
.masterid"]},"operator":"FIELD"]},"operands":[],"active":true,"attributes":"","value":[],"operator":"NUL
L"}]},"active":true,"attributes":"","value":[],"operator":"is
not"}],"active":true,"attributes":"","value":[],"operator":"and"},"metadata":""}</Filter>
          <GroupingFields>brand</GroupingFields>
          <ReferenceItemAlgorithmLink>default-use-first-item-
algorithm</ReferenceItemAlgorithmLink>
          <Active>true</Active>
        </AttributeAnalyzer>
      </AttributeAnalyzers>
    </AttributeFamily>
  </FamilyRelations>

</ProductRelationImport>
```

XML Tag	Format	Example	Mandatory	Default Value	Description
Manual Family			x		
Manual Group			x		
ID	String	ManualGroup_1	x		Description
ReferenceItem	String	MR318412	x		Item for Price calculation
Item	String	MR31841220	x		Item ID
AttributeFamily					
AttributeAnalyzers					
AttributeAnalyzer					
ID	String	AttributeAnalyzer_12345	x		
Name	String	T-Shirts	x		
Description	String	Same Price T-Shirts			Description
Mergable	Boolean	true	x	false	Grouping logic can / cannot be changed or extended by subsequent grouping logic
GroupingFields	String	brand			Product Attributes for grouping
ReferenceItemAlgorithmLink	String	default-use-first-item-algorithm			Strategy for picking a reference item Supported <ul style="list-style-type: none"> "default-use-first-item-algorithm" Use first item in group as reference item, aggregate stock from group members to reference item Use your custom ID if you implemented a custom strategy
AggregationAlgorithmLink	String	default-aggregation-strategy			Strategy for aggregating dependent item data to reference item. Supported <ul style="list-style-type: none"> "default-aggregation-strategy" Aggregate stock from group members to reference item Use your custom ID if you implemented a custom strategy
Active	Boolean	true	x	true	Is Analyser for Grouping enabled or disabled

Delete Families

Delete specific manual family

For manual family members you can delete a single family this way:

```

<ProductRelationImport Version="1.0.0">
  <FamilyRelations>
    <ManualFamily/>
    <Families>
      <Family Delete="true">
        <ReferenceItem></ReferenceItem>
      </Family>
    </Families>
  </ManualFamily>
</FamilyRelations>
</ProductRelationImport>

```

Delete all families of a type

You can delete all families of a type at once.


```

<ProductRelationImport Version="1.0.0">

  <FamilyRelations>

    <ManualFamily Delete="true"/>
  </ManualFamily>

    <AttributeFamily Delete="true">
      <Id></Id>
    </AttributeFamily>

    <CustomFamily Type="" Delete="true">
      <Id></Id>
    </CustomFamily>

  </FamilyRelations>

</ProductRelationImport>

```

Mapping: Quantity Relations

```

<ProductRelationImport Version="1.0.0">

  <QuantityRelations>
    <QuantityBasedAnalyzer>
      <ID>QuantityBasedAnalyzer_12345</ID>
      <Name>Test Attribute Update</Name>
      <Description>Test Description</Description>
      <Mergeable>true</Mergeable>
      <Filter></Filter>
      <GroupingFields>brand</GroupingFields>
      <ReferenceItemAlgorithmLink>lowest-quantity-reference-item-
algorithm</ReferenceItemAlgorithmLink>
      <Active>true</Active>
      <RelationPercentage>2</RelationPercentage>
      <PriceMode>EXACT</PriceMode>
      <RelationType>LINEAR</RelationType>
      <ReferenceItemMeasureUnitCodes>PACK</ReferenceItemMeasureUnitCodes>
    </QuantityBasedAnalyzer>
  </QuantityRelations>

</ProductRelationImport>

```

XML Tag	Format	Example	Mandatory	Default	Description
QuantityRelations			x		
QuantityBasedAnalyzer					
ID	String	QuantityBasedAnalyzer_12345	x		ID
Name	String	Test Attribute Update	x		Name of the group
Description	String	Test Description			Description of the group
Mergeable	BOOL	True/False	x	false	Grouping logic can / cannot be changed or extended by subsequent grouping logic
GroupingFields	String	brand	x		Product Attributes for grouping
ReferenceItemAlgorithmLink	String	lowest-quantity-reference-item-algorithm	x		Strategy for reference item Supported <ul style="list-style-type: none"> lowest-quantity-reference-item-algorithm Take item with lowest quantity value in base unit measure-unit-code-reference-item-algorithm Take the item having the measure unit code from a given list
Active	BOOL	True/False	x		Is Analyser for Grouping enabled or disabled
RelationPercentage	INT	2	x		Value for price distance in percent
PriceMode	String	EXACT	x		Sets the mode of relation rule for price calculation Supported <ul style="list-style-type: none"> EXACT Exact relation MINIMUMDISTANCE Define minimum distance between items
RelationType	String	LINEAR/HIGHERPRICE/LOWERPRICE	x		Sets type of relation
ReferenceItemMeasureUnitCodes	String comma separated	PACK			Comma separated list of measure unit codes that should be used for finding the reference item. E.g. you could put there PIECE. So the item with measureUnitCode=PIECE would be used as reference item. Only works if you set for ReferenceItemAlgorithmLink: measure-unit-code-reference-item-algorithm

Delete all quantity relations of a type

You can delete all families of a type at once.

```

<ProductRelationImport Version="1.0.0">

  <QuantityRelations>

    <QuantityRelation Delete="true">
      <id></Id>
    </QuantityRelation>

  </QuantityRelations>

</ProductRelationImport>

```

Mapping: Item Link Relations

```

<ProductRelationImport Version="1.0.0">

  <ItemLinkRelations>
    <ItemLinkRelation>
      <ID>ItemLinkRelation_1</ID>
      <Name>Rel1</Name>
      <ItemLinkRelationRule>
        <ItemLinkRelationGroups>
          <ItemLinkRelationGroup>
            <Item>MR31841220</Item>
          </ItemLinkRelationGroup>
          <ItemLinkRelationGroup>
            <Item>MR31841230</Item>
            <Item>MR31841240</Item>
          </ItemLinkRelationGroup>
        </ItemLinkRelationGroups>
        <ItemLinkRelationConditions>
          <ItemLinkRelationCondition>
            <Offset>10</Offset>
            <Offsetmessuretype>ABSOLUTE</Offsetmessuretype>
            <ConvertPackaging>false</ConvertPackaging>
            <Offsettype>MINIMAL</Offsettype>
            <Type>CHEAPER</Type>
          </ItemLinkRelationCondition>
        </ItemLinkRelationConditions>
      </ItemLinkRelationRule>
    </ItemLinkRelation>
  </ItemLinkRelations>

</ProductRelationImport>

```

XML Tag	Format	Example	Mandatory	Default	Description
ItemLinkRelations			x		
ItemLinkRelation					
ID	String	ItemLinkRelation_1	x		ID
Name	String	Rel1	x		Name of the group
ItemLinkRelationRule					
ItemLinkRelationGroups					
ItemLinkRelationGroup					
Item	String	MR31841220	x		Item ID
ItemLinkRelationConditions					
ItemLinkRelationCondition					
Offset	INT	10	x		Value for Relationgroup offset
Offsetmessuretype	String	ABSOLUTE	x		ABSOLUTE or PERCENTAGE
ConvertPackaging	Boolean	false	x		Enable / Disable conversion of packaging size
Offsettype	String	MINIMAL	x		Supported <ul style="list-style-type: none"> MINIMAL Minimal Distance is defined MAXIMAL Maximal Distance is defined
Type	String	CHEAPER	x		Supported <ul style="list-style-type: none"> CHEAPER Item should be cheaper MAXIMAL Item should be more expensive

Delete Item Relations

Delete specific item relation

You can delete a single item relation this way:

```
<ProductRelationImport Version="1.0.0">
  <ItemLinkRelations>
    <ItemLinkRelationGroup Delete="true">
      <Id></Id>
    </ItemLinkRelationGroup>
  </ItemLinkRelations>
</ProductRelationImport>
```

Delete all item relations

You can delete all item relations at once.

```
<ProductRelationImport Version="1.0.0">
  <ItemLinkRelations Delete=true/>
</ProductRelationImport>
```

2.5.13.5 Configuration

General

```
<importMode>INITIAL</importMode>
```

Tag	Format	Mandatory	Default	Example	Description
importMode	INITIAL UPDATE		UPDATE		<ul style="list-style-type: none"> INITIAL: Delete all current product relations before importing new ones UPDATE: Updates the product relations

Communication

Both communication paths can be configured. If configured, the HTTP endpoint will always have precedence during import.

Import via File System

See "Import Communication Interface".

Import via HTTP Endpoint

For general description, see "Import Communication Interface".

XML Reading

Please see details about the configuration [here](#).

CONTACT

GK Software SE
Waldstraße 7
08261 Schöneck
Germany

T +49 (0) 3 74 64 84 – 0
F +49 (0) 3 74 64 84 – 15
documentation@gk-software.com
www.gk-software.com