PUBLIC
SAP HANA Smart Data Integration and SAP HANA Smart Data Quality 2.0 SP03
Document Version: 1.0 – 2026-08-26

Administration Guide

© 2026 SAP SE or an SAP affiliate company. All rights reserved.

Content

1 Administration Guide for SAP HANA Smart Data Integration and SAP HANA Smart Data Quality. 8

2 Monitoring Data Provisioning in the SAP HANA Database Explorer. 9

2.1 Access the Data Provisioning Monitors. 9

2.2 Monitoring Data Provisioning Agents and Agent Groups. 10

Information Available on the Data Provisioning Agent Monitors. 11

2.3 Monitoring Remote Sources. 12

2.4 Monitoring Remote Subscriptions. 13

2.5 Monitoring Data Provisioning Tasks. 15

Information Available on the Data Provisioning Task Monitor. 16

3 Monitoring Data Provisioning in the SAP HANA Web-based Development Workbench. 20

3.1 Enable Statistics Collection. 20

3.2 Access the Data Provisioning Monitors. 22

3.3 Common Concepts and Controls. 23

3.4 Monitoring Agents. 26

3.5 Monitoring Remote Sources. 28

Measuring Latency. 32

Measuring Throughput. 34

Measuring Record Volume. 36

3.6 Monitoring Remote Subscriptions. 37

Remote Subscription Statuses. 39

3.7 Monitoring Remote Queries. 40

3.8 Monitoring Design-Time Objects. 44

3.9 Monitoring Tasks. 45

Change Retention Period for Data Provisioning Task Monitor. 46

3.10 Monitoring Statistics Reference. 48

3.11 Configure User Settings Profiles. 119

3.12 Create Notifications. 120

3.13 Creating Monitoring Alerts. 121

4 Creating and Monitoring Data Assurance Jobs. 123

4.1 Data Assurance Job Options. 125

4.2 Data Assurance Job Monitor Statistics. 130

4.3 Data Assurance Framework Configuration. 131

4.4 SAP HANA (Remote) to SAP HANA (Target) Data Type Mapping With Data Assurance Adapter 133

4.5 Comparison and Reconciliation Strategies. 134
4.6 Data Assurance Objects Created Upon Initialization. 136
4.7 Creating a Data Assurance Subscription that Contains a Computed Column Expression. 138

5 Administering Data Provisioning. 139
5.1 Managing Agents and Adapters. 139
Manage Agents from the Data Provisioning Agent Monitor. 140
Manage Adapters from the Data Provisioning Agent Monitor. 142
Manage Agents Remotely. 143
Back Up the Data Provisioning Agent Configuration. 145
Uninstall the Data Provisioning Agent. 145
5.2 Managing Agent Groups. 146
Failover Behavior in an Agent Group. 147
Load Balancing in an Agent Group. 148
Create or Remove an Agent Group. 148
Manage Agent Nodes in an Agent Group. 150
Add Adapters to an Agent Group. 153
Configure Remote Sources in an Agent Group. 154
5.3 Managing Remote Sources and Subscriptions. 157
Create a Remote Source. 158
Suspend and Resume Remote Sources. 162
Alter Remote Source Parameters. 163
Manage Remote Subscriptions. 164
Processing Remote Source or Remote Subscription Exceptions. 165
5.4 Managing Design Time Objects. 165
Execute Flowgraphs and Replication Tasks. 166
Schedule Flowgraphs and Replication Tasks. 167
Stop Non-Realtime Flowgraph Executions. 169
Start and Stop Data Provisioning Tasks. 170
Schedule Data Provisioning Tasks. 171
Executing Partitions. 172
5.5 Managing Enterprise Semantic Services 174
Roles for Enterprise Semantic Services. 175
Enterprise Semantic Services Knowledge Graph and Publication Requests. 176
Publishing Artifacts. 177
Monitor the Status of Publication Requests. 180
Manage Published Artifacts. 183
Data Profiling. 186
Setting Configuration Parameters. 188
Troubleshooting Enterprise Semantic Services. 189

6 Maintaining Connected Systems. 199

6.1 Maintaining the SAP HANA System. . . . 199
Update the SAP HANA System. . . . 200
Takeover/Failback with SAP HANA System Replication. . . . 200
Failover with SAP HANA Scale-Out. . . . 201
Migrate Objects with SAP HANA System Copy. . . . 202
6.2 Maintaining Source Databases. . . . 204
Restart the Source Database. . . . 205
Change the Source Database User Password. . . . 205
Cleaning up LogReader Archives. . . . 206
Cleaning up PostgreSQL Source Objects. . . . 209
Recover from Missing LogReader Archives. . . . 209
Recover from LogReader Database Upgrades. . . . 210
Change the Primary Archive Log Path During Replication. . . . 211
Maintain the Source Database Without Propagating Changes to SAP HANA. . . . 212
Recover with Microsoft SQL Server Always On Failover. . . . 213
Recover with SAP HANA System Replication Failover. . . . 213

7 Troubleshooting and Recovery Operations. . . . 215
7.1 Troubleshooting Real-Time Replication Initial Queue Failures. . . . 215
Resolve User Privilege Errors. . . . 216
Resolve Remote Source Parameter Errors. . . . 217
Resolve Improper Source Database Configuration. . . . 218
Resolve Improper Adapter Configurations on the Agent. . . . 228
Resolve Uncommitted Source Database Transactions. . . . 230
Resolve Log Reader Instance Port Conflicts. . . . 231
Resolve Data Provisioning Server Timeouts. . . . 232
Load Clustered and Pooled Table Metadata into SAP HANA. . . . 233
7.2 Recovering from Replication Failures. . . . 233
Check for Log Reader Errors. . . . 234
Recover from a Source Table DDL Schema Change. . . . 235
Recover from a Truncated Source Table. . . . 236
Recover from Source Table and Replication Task Recreation. . . . 236
Recover from a Source and Target Data Mismatch. . . . 237
Recover from Data Inconsistencies. . . . 238
Recover from an Agent Communication Issue. . . . 240
Resolve Stopped or Delayed Replication on Oracle. . . . 241
Resolve Locked SAP HANA Source Tables. . . . 242
Reset the Remote Subscription. . . . 242
Clear Remote Subscription Exceptions. . . . 244
7.3 Recovering from Crashes and Unplanned System Outages. . . . 244
Recover from an Index Server Crash. . . . 245
Recover from a Data Provisioning Server Crash. . . . 245

Recover from a Data Provisioning Agent JVM Crash. . . . 246
Recover from an Unplanned Source Database Outage. . . . 247
Recover from an SAP ASE Adapter Factory Crash. . . . 248
7.4 Troubleshooting Data Provisioning Agent Issues. . . . 249
Data Provisioning Agent Log Files and Scripts. . . . 249
Clean an Agent Started by the Root User. . . . 250
Agent JVM Out of Memory. . . . 251
Adapter Prefetch Times Out. . . . 252
Uninstalled Agent Reports Alerts or Exceptions. . . . 252
Create an Agent System Dump. . . . 253
Resolve Agent Parameters that Exceed JVM Capabilities. . . . 254
7.5 Troubleshooting Other Issues. . . . 255
Activate Additional Trace Logging for the Data Provisioning Server. . . . 256
Resolve a Source and Target Data Mismatch. . . . 258
Configuring the Operation Cache. . . . 258
Ensure Workload Management and Resource Consumption. . . . 260

8 SQL and System Views Reference for Smart Data Integration and Smart Data Quality. . . . 261
8.1 SQL Functions. . . . 261
SESSION_CONTEXT Function [Smart Data Integration]. . . . 262
8.2 SQL Statements. . . . 262
ALTER ADAPTER Statement [Smart Data Integration]. . . . 264
ALTER AGENT Statement [Smart Data Integration]. . . . 265
ALTER REMOTE SOURCE Statement [Smart Data Integration]. . . . 267
ALTER REMOTE SUBSCRIPTION Statement [Smart Data Integration]. . . . 271
CANCEL TASK Statement [Smart Data Integration]. . . . 272
CREATE ADAPTER Statement [Smart Data Integration]. . . . 274
CREATE AGENT Statement [Smart Data Integration]. . . . 276
CREATE AGENT GROUP Statement [Smart Data Integration]. . . . 278
CREATE AUDIT POLICY Statement [Smart Data Integration]. . . . 279
CREATE REMOTE SOURCE Statement [Smart Data Integration]. . . . 281
CREATE REMOTE SUBSCRIPTION Statement [Smart Data Integration]. . . . 282
CREATE VIRTUAL PROCEDURE Statement [Smart Data Integration]. . . . 288
DROP ADAPTER Statement [Smart Data Integration]. . . . 290
DROP AGENT Statement [Smart Data Integration]. . . . 291
DROP AGENT GROUP Statement [Smart Data Integration]. . . . 293
DROP REMOTE SUBSCRIPTION Statement [Smart Data Integration]. . . . 294
GRANT Statement [Smart Data Integration]. . . . 295
PROCESS REMOTE SUBSCRIPTION EXCEPTION Statement [Smart Data Integration]. . . . 297
START TASK Statement [Smart Data Integration]. . . . 298
8.3 System Views. . . . 302
ADAPTER_CAPABILITIES System View [Smart Data Integration]. . . . 305

ADAPTER_LOCATIONS System View [Smart Data Integration]. 305
ADAPTERS System View [Smart Data Integration]. 306
AGENT_CONFIGURATION System View [Smart Data Integration]. 306
AGENT_GROUPS System View [Smart Data Integration]. 307
AGENTS System View [Smart Data Integration]. 307
M_AGENTS System View [Smart Data Integration]. 308
M_REMOTE_SOURCES System View [Smart Data Integration]. 308
M_REMOTE_SUBSCRIPTION_COMPONENTS System View [Smart Data Integration]. 309
M_REMOTE_SUBSCRIPTION_STATISTICS System View [Smart Data Integration]. 310
M_REMOTE_SUBSCRIPTIONS System View [Smart Data Integration]. 310
M_REMOTE_QUERY_STATISTICS System View [Smart Data Integration]. 312
M_SESSION_CONTEXT System View [Smart Data Integration]. 314
REMOTE_SOURCE_OBJECT_COLUMNS System View [Smart Data Integration]. 315
REMOTE_SOURCE_ OBJECT_DESCRIPTIONS System View [Smart Data Integration]. 316
REMOTE_SOURCE_OBJECTS System View [Smart Data Integration]. 316
REMOTE_SOURCES System View [Smart Data Integration]. 317
REMOTE_SUBSCRIPTION_EXCEPTIONS System View [Smart Data Integration]. 317
REMOTE_SUBSCRIPTIONS System View [Smart Data Integration]. 318
TASK_CLIENT_MAPPING System View [Smart Data Integration]. 319
TASK_COLUMN_DEFINITIONS System View [Smart Data Integration]. 319
TASK_EXECUTIONS System View [Smart Data Integration]. 320
TASK_LOCALIZATION System View [Smart Data Integration]. 321
TASK_OPERATIONS System View [Smart Data Integration]. 321
TASK_OPERATIONS_EXECUTIONS System View [Smart Data Integration]. 322
TASK_PARAMETERS System View [Smart Data Integration]. 323
TASK_TABLE_DEFINITIONS System View [Smart Data Integration]. 323
TASK_TABLE_RELATIONSHIPS System View [Smart Data Integration]. 324
TASKS System View [Smart Data Integration]. 325
VIRTUAL_COLUMN_PROPERTIES System View [Smart Data Integration]. 326
VIRTUAL_TABLE_PROPERTIES System View [Smart Data Integration]. 326
BEST_RECORD_GROUP_MASTER_STATISTICS System View [Smart Data Quality]. 327
BEST_RECORD_RESULTS System View [Smart Data Quality]. 328
BEST_RECORD_STRATEGIES System View [Smart Data Quality]. 329
CLEANSE_ADDRESS_RECORD_INFO System View [Smart Data Quality]. 329
CLEANSE_CHANGE_INFO System View [Smart Data Quality]. 330
CLEANSE_COMPONENT_INFO System View [Smart Data Quality]. 331
CLEANSE_INFO_CODES System View [Smart Data Quality]. 333
CLEANSE_STATISTICS System View [Smart Data Quality]. 334
GEOCODE_INFO_CODES System View [Smart Data Quality]. 334
GEOCODE_STATISTICS System View [Smart Data Quality]. 335
MATCH_GROUP_INFO System View [Smart Data Quality]. 336

MATCH_RECORD_INFO System View [Smart Data Quality]. 336
MATCH_SOURCE_STATISTICS System View [Smart Data Quality]. 337
MATCH_STATISTICS System View [Smart Data Quality]. 338
MATCH_TRACING System View [Smart Data Quality]. 339

1 Administration Guide for SAP HANA Smart Data Integration and SAP HANA Smart Data Quality

This guide describes the common tasks and concepts necessary for the ongoing operation, administration, and monitoring of SAP HANA smart data integration and SAP HANA smart data quality.

The following areas are covered:

- Monitoring
- Administration and maintenance tasks
- Troubleshooting and recovery operations

For information about the initial installation and configuration of SAP HANA smart data integration and SAP HANA smart data quality, refer to the *Installation and Configuration Guide for SAP HANA Smart Data Integration and SAP HANA Smart Data Quality*.

For information about administration of the overall SAP HANA system, refer to the *SAP HANA Administration Guide*.

Related Information

Configuration Guide for Other SAP HANA Scenarios
SAP HANA Administration Guide

2 Monitoring Data Provisioning in the SAP HANA Database Explorer

Monitors let you view status and other key performance indicators for agents, remotes sources, and tasks, for example.

In the SAP HANA database explorer, you can view status and other monitoring information for agents, remote sources and subscriptions, tasks, and design-time objects (flowgraphs and replication tasks) created in an SAP HANA system.

Access the Data Provisioning Monitors [page 9]
You can view agents, remote sources and subscriptions, and tasks created in an SAP HANA system from the SAP HANA cockpit or from within the SAP HANA database explorer.

Monitoring Data Provisioning Agents and Agent Groups [page 10]
Within the SAP HANA database explorer, you can monitor basic system information of an agent such as status, memory, and the time it last connected with the Data Provisioning Server.

Monitoring Remote Sources [page 12]
The SAP HANA database explorer provides information about your remote sources.

Monitoring Remote Subscriptions [page 13]
The SAP HANA database explorer provides information about your remote subscriptions.

Monitoring Data Provisioning Tasks [page 15]
The SAP HANA database explorer provides information about your replication tasks and transformation tasks.

Related Information

2.1 Access the Data Provisioning Monitors

You can view agents, remote sources and subscriptions, and tasks created in an SAP HANA system from the SAP HANA cockpit or from within the SAP HANA database explorer.

Prerequisites

Your web browser must support the SAPUI5 library `sap.m` (for example, Microsoft Internet Explorer 9).

For more information about SAPUI5 browser support, see SAP Note 1716423 and the *Product Availability Matrix (PAM) for SAPUI5*.

Context

You can view these monitors in the following ways:

- From the SAP HANA cockpit
- From within the SAP HANA database explorer

Procedure

- To view monitors from the SAP HANA cockpit:
  a. Open the SAP HANAcockpit in your browser.
  b. If necessary, enter your database user name and password.
  c. Go to the smart data integration block of links in the *System Monitor* page.
- To view monitors from within the SAP HANA database explorer:
  a. In SAP Web IDE, choose *Tools* > *Database Explorer* .
  b. Choose your system and expand the *Catalog* folder.
  c. To view a list of objects, select a Catalog item and the list displays in the pane below. Select an object in the pane to see a detail page.
  d. To view monitoring data of a Category item, right-click the item and select *Show <items>*.

Task overview: Monitoring Data Provisioning in the SAP HANA Database Explorer [page 9]

Related Information

Monitoring Data Provisioning Agents and Agent Groups [page 10]
Monitoring Remote Sources [page 12]
Monitoring Remote Subscriptions [page 13]
Monitoring Data Provisioning Tasks [page 15]

2.2 Monitoring Data Provisioning Agents and Agent Groups

Within the SAP HANA database explorer, you can monitor basic system information of an agent such as status, memory, and the time it last connected with the Data Provisioning Server.

You can sort and hide individual columns by right-clicking a row and selecting your display preferences.

Parent topic: Monitoring Data Provisioning in the SAP HANA Database Explorer [page 9]

Related Information

Access the Data Provisioning Monitors [page 9]
Monitoring Remote Sources [page 12]
Monitoring Remote Subscriptions [page 13]
Monitoring Data Provisioning Tasks [page 15]

Information Available on the Data Provisioning Agent Monitors

To see an overview of all agents, in the database explorer expand the database's *Catalog* folder, right-click *Agents*, and select *Show Agents*.

Table 1: Information Available in the Agent Overview Table

Column	Description
Agent Name	Name of the Data Provisioning Agent.
Agent Host	Name of the host on which the agent is running.
Agent Port	Port that the agent uses to communicate with the Data Provisioning Server.
Agent Group	Group to which the agent has been assigned
Status	State of the agent. The following states are possible: • CONNECTING • DISCONNECTED • CONNECTED
Since Last Connect	Elapsed time since the last connection from the Data Provisioning Server to the Data Provisioning Agent.
Last Connect Time	The last connect time from the Data Provisioning Server to the Data Provisioning Agent.
Adapters	Number of adapters defined for this Data Provisioning Agent.
Protocol	Type of network protocol used between the Data Provisioning Agent and the Data Provisioning Server. The following protocols are possible: • TCP • HTTP
Used Memory	Amount of memory currently used by the agent.
Used Swap Space	Amount of swap space currently used by the agent.
Free Memory	Amount of free memory on the agent host.
Free Swap Space	Amount of free swap space on the agent host.

Column	Description
Is SSL Enabled	Specifies whether the agent listening on TCP port uses SSL

To see an overview of agent groups, in the database explorer expand the database's *Catalog* folder, right-click *Agent Groups*, and select *Show Agent Groups*.

Table 2: Information Available in the Agent Groups Table

Column	Description
Agent Group	Name of the Data Provisioning Agent group.
Agent Count	Number of agents added to this Agent Group
Connected Agent Count	Number of connected agents in this Agent Group

2.3 Monitoring Remote Sources

The SAP HANA database explorer provides information about your remote sources.

Parent topic: Monitoring Data Provisioning in the SAP HANA Database Explorer [page 9]

Related Information

Access the Data Provisioning Monitors [page 9]
Monitoring Data Provisioning Agents and Agent Groups [page 10]
Monitoring Remote Subscriptions [page 13]
Monitoring Data Provisioning Tasks [page 15]

2.3.1 Information Available on the Data Provisioning Remote Sources Monitor

To see an overview of all remote sources, in the database explorer expand the database's *Catalog* folder, right-click *Remote Sources*, and select *Show Remote Sources*.

Table 3: Information Available in Remote Sources Overview Table

Column	Description
Remote Source Name	Name of the remote source.
Adapter Name	Name of the adapter.
Location	Port that the agent uses to communicate with the Data Provisioning Server.

Column	Description
Agent Name	Name of the agent for this remote source.
Status	The status of changed-data capture on this remote source. The following status values are possible: • OK • ERROR • SUSPENDED
Total Subscriptions	Number of subscriptions defined for this remote source.

To see details for a remote source, in the database explorer expand the database's *Catalog* folder and select *Remote Sources*. From the resulting list of tasks in the pane below, select a task.

The detail page displays information in four sections: *Remote Objects*, *Dictionary*, *Statistics*, and *Exceptions*.

You can click *Edit* to edit the properties of the remote source.

2.4 Monitoring Remote Subscriptions

The SAP HANA database explorer provides information about your remote subscriptions.

Parent topic: Monitoring Data Provisioning in the SAP HANA Database Explorer [page 9]

Related Information

Access the Data Provisioning Monitors [page 9]
Monitoring Data Provisioning Agents and Agent Groups [page 10]
Monitoring Remote Sources [page 12]
Monitoring Data Provisioning Tasks [page 15]

2.4.1 Information Available on the Data Provisioning Remote Subscriptions Monitor

To see an overview of all remote subscriptions, in the database explorer expand the database's *Catalog* folder, right-click *Remote Subscriptions*, and select *Show Remote Subscriptions*.

Remote Subscription Actions

You can start an ALTER REMOTE SUBSCRIPTION statement with these commands: Queue, Distribute, and Reset. You can drop the subscription with the DROP REMOTE SUBSCRIPTION statement. Select one or more remote subscriptions, and click one of the buttons.

Command	Description
Queue	Initiate real-time data processing. Typically, the initial load of data is preceded by the Queue command.
Distribute	Applies the changes after the initial load completes.
Reset	Restarts the real-time process from the initial load.
Drop	Removes the remote source.

Table 4: Information Available in Remote Subscriptions Table

Column	Description
Subscription Name	Name of the remote subscription.
Schema Name	Name of the schema.
Remote Source Name	Name of the remote source for which this subscription is defined.
Valid	Whether or not the remote subscription is valid.
Subscription State	Name of the state of the remote subscription. For more information, see Remote Subscription Statuses [page 39].
Last Processed	Elapsed time since the last changed data was processed.
Last Processed Transaction Time	Time the last changed data was processed.
Subscription Type	Type of subscription. The following values are possible: • TABLE • VIRTUAL TABLE
Target Type	Type of target. The following values are possible: • TABLE • VIRTUAL TABLE

To see details for a remote subscription, in the database explorer expand the database's *Catalog* folder and select *Remote Subscriptions*. From the resulting list of subscriptions in the pane below, select a subscription.

Table 5: Information Available in Remote Subscription Statistics Table

Column	Description
Schema Name	Name of the schema (user name) in the remote source.
Remote Source Name	Name of the remote source.
Subscription Name	Name of the remote subscription.
Received Count	Total number of messages received by the Data Provisioning Server.
Applied Count	Total number of messages applied.
Received Size	Total size of messages received by the Data Provisioning Server.
Applied Size	Total size of messages applied.
Rejected Count	Total number of messages rejected.
Since Last Message Received	Time elapsed between now and when the last message was received.
Last Message Received Time	Time the last message was received.
Since Last Message Applied	Time elapsed between now and when the last message was applied.
Last Message Applied	Time the last message was applied.

2.5 Monitoring Data Provisioning Tasks

The SAP HANA database explorer provides information about your replication tasks and transformation tasks.

You can sort and hide individual columns by right-clicking a row and selecting your display preferences. You can also start and stop tasks.

Execute Tasks

From the *Task Overview* or the *Task Details* page, select a task that you want to run. You can select to run more than one task on the *Task Overview* page. Click *Execute*.

Cancel Tasks

From the *Task Execution* table in the *Task Details* page, select a running task. You can select multiple tasks from the *Task Execution* table. Click *Cancel Execution*.

Parent topic: Monitoring Data Provisioning in the SAP HANA Database Explorer [page 9]

Related Information

Access the Data Provisioning Monitors [page 9]
Monitoring Data Provisioning Agents and Agent Groups [page 10]
Monitoring Remote Sources [page 12]
Monitoring Remote Subscriptions [page 13]
Executing Partitions [page 172]

Information Available on the Data Provisioning Task Monitor

To see an overview of all tasks, in the database explorer expand the database's *Catalog* folder, right-click *Tasks*, and select *Show Tasks*.

Table 6: Information Available in the Task Overview Table

Column	Description
Comments	Description of the task, from the task plan.
Create Time	The time that the task was created.
Has Table Type Input	TRUE if the task is modeled with a table type as input. This means data would need to be passed (pushed) at execution time.
Has SDQ	TRUE if the task contains smart data quality (SDQ) functionality.
Has Variables	TRUE if the task contains variables.
Input Parameter Count	Number of input (tableType) parameters.
Is read-only	TRUE if the task is read only (has only table type outputs), FALSE if it writes to non-table-type outputs.
Is valid	TRUE if the task is in a valid state, FALSE if it has been invalidated by a dependency.
Memory Size	Memory size of loaded task.
Plan Version	Version of the task plan
Procedure Name	If the task was created with a procedure instead of a plan, this attribute contains the name of the stored procedure.
Procedure Schema	If the task was created with a procedure instead of a plan, this attribute contains the schema name of the stored procedure.
Realtime	TRUE if the task is a real-time task, else FALSE.
Realtime Design Time Object	TRUE if the task is a real-time design time object like a replication or flowgraph task.
Owner Name	Owner of the task.
Output Parameter Count	Number of output (tableType) parameters.
Schema Name	Name of the schema in which the task was created.
SQL Security	Security model for the task, either DEFINER or INVOKER.
Task ID	Unique identifier for a task.

Column	Description
Task Name	Name of the data provisioning task.
Task Type	Type of task. Derived from task plan.

To see details for a task, in the database explorer expand the database's *Catalog* folder and select *Tasks*. From the resulting list of tasks in the pane below, select a task.

The detail page displays information in three sections: *Task Executions*, *Task Partitions*, and *Task Operations*. If you select a task in the first table, the other tables show information for only that task.

Use the row-count display to change the number of task executions to display at a time in each table. The default is 500. After selecting a different task, refresh the tables to see all corresponding information.

Table 7: Information Available in Task Executions Table

Column	Description
Task Name	Name of the Data Provisioning task.
Schema Name	Name of the schema in which the task was created.
Host	Name of the host on which the task is running.
Port	Port number that the task uses to communicate with the Data Provisioning Server.
Task Execution ID	Unique identifier for the task.
Partition Count	Number of logical partitions used for parallel processing in the task. If no partitions are defined for the task, the partition count is 1.
Start Time	Day, date, and time when the task started.
End Time	Day, date, and time when the task ended.
Duration	Total elapsed time from start to end for COMPLETED or FAILED tasks. Current elapsed time for RUNNING tasks.
Status	Current status of the task. The following values are possible: • STARTING • RUNNING • COMPLETED • FAILED • CANCELLED (or, CANCELLING)
Total Progress	Percentage completed.
Processed Records	Number of records that the task has processed.
Async	Indicates whether or not the task is running as a background task. Possible values are: • TRUE • FALSE

Column	Description
Parameters	Parameters passed to the task.
Has Remote Statements	Indicates whether there's one or more remote statements.
Connection ID	Connection identifier
Transaction ID	Transaction identifier used for the task execution
HANA User	Name of the user that started the execution of this task.
Application User	Application user set in the session context.

Table 8: Information Available in Task Partitions Table

Column	Description
Task Name	Name of the Data Provisioning task.
Schema Name	Name of the schema in which the task was created.
Task Execution ID	Unique identifier for the task.
Partition Name	Name of the partition.
Partition ID	Identification number of the logical partition used for parallel processing within the task.
Start Time	Day, date, and time when the task started.
End Time	Day, date, and time when the task ended.
Duration	Total elapsed time from start to end for COMPLETED or FAILED tasks. Current elapsed time for RUNNING tasks.
Status	Current status of the task. The following values are possible: • STARTING • RUNNING • COMPLETED • FAILED • CANCELLED (or, CANCELLING)
Total Progress	Percentage completed.
Processed Records	Number of records that the task has processed.
Has Remote Statements	Indicates whether there's one or more remote statements.
Connection ID	Connection identifier
Transaction ID	Transaction identifier used for the task execution

Table 9: Information Available in Task Operations Table

Column	Description
Operation	Name of the operation that the task is currently performing. For a transformation task, the name of the operation is the name of the node that appears in the flowgraph.

Column	Description
Schema Name	Name of the schema (user name) in the remote source.
Task Name	Name of the task.
Task Execution ID	Identification number of the task.
Partition ID	Identification number of the logical partition used for parallel processing within the task.
Partition Name	Name of the logical partition used for parallel processing within the task.
Operation Type	Current type of operation. For example, the operation type can be Table Writer, Adapter, Projection, and so forth.
Start Time	Day, date, and time when the task started.
End Time	Day, date, and time when the task ended.
Duration	Total elapsed time from start to end.
Status	Current status of the task. The following values are possible: • STARTING • RUNNING • COMPLETED • FAILED
Progress	Percentage completed.
Processed Records	Number of records that the task has processed.
Side Effects	Indicates whether or not this operation generates side effect statistics. Possible values are: • TRUE • FALSE

3 Monitoring Data Provisioning in the SAP HANA Web-based Development Workbench

The Data Provisioning monitors are browser-based interfaces that let you monitor agents, remote sources and subscriptions, design-time objects (flowgraphs and replication tasks), and tasks created in an SAP HANA system.

Related Information

Enable Statistics Collection [page 20]
Access the Data Provisioning Monitors [page 22]
Common Concepts and Controls [page 23]
Monitoring Agents [page 26]
Monitoring Remote Sources [page 28]
Monitoring Remote Subscriptions [page 37]
Monitoring Remote Queries [page 40]
Monitoring Design-Time Objects [page 44]
Monitoring Tasks [page 45]
Monitoring Statistics Reference [page 48]
Configure User Settings Profiles [page 119]
Create Notifications [page 120]
Creating Monitoring Alerts [page 121]

3.1 Enable Statistics Collection

For some versions of SAP HANA, to enable statistics collection for the monitors you must activate the dpStatistics job.

Prerequisites

If you are using one of the following SAP HANA versions, enable statistics collection per the following procedure. For more information, see the *SAP HANA smart data integration Product Availability Matrix (PAM)*.

- SAP HANA 1.0 SPS 12

- SAP HANA 2.0 SPS 03: versions prior to 2.00.037.04
- SAP HANA 2.0 SPS 04: versions prior to 2.00.045

The user must have the following roles or privileges.

Table 10: Roles and Privileges

Action	Role or Privilege
Enable statistics collection	Role: sap.hana.xs.admin.roles::JobSchedulerAdministrator Execute privilege on the procedure: "SAP_HANA_IM_DP"."sap.hana.im.dp.monitor.ds::LOG_DP_STATISTICS"

Context

Activate the dpStatistics job as follows.

Procedure

1. Launch the XS Job Admin Dashboard `/sap/hana/xs/admin/jobs/`.
2. For the sap.hana.im.dp.monitor.jobs::dpStatistics job, on the *XS Job Details* page /sap/hana/xs/admin/jobs/#/package/sap.hana.im.dp.monitor.jobs/job/dpStatistics:
   a. On the *Configuration* tab, enter the *User* and *Password*.
   b. Select the *Active* check box
   c. Select *Save Job*.

Related Information

Assign Roles and Privileges

SAP HANA smart data integration and all its patches Product Availability Matrix (PAM) for SAP HANA SDI 2.0

3.2 Access the Data Provisioning Monitors

In the SAP HANA Web-based Development Workbench, there are several ways to access the Data Provisioning monitors.

Prerequisites

- You must have the following roles or privileges to use the data provisioning monitors:

  Table 11: Roles and Privileges

Action	Role or Privilege
Access the monitors	Role: sap.hana.im.dp.monitor.roles::Monitoring
Access the monitors and perform administration tasks	Role: sap.hana.im.dp.monitor.roles::Operations

- Your web browser must support the SAPUI5 library `sap.m` (for example, Microsoft Internet Explorer 9). For more information about SAPUI5 browser support, see SAP Note 1716423 and the Product Availability Matrix (PAM) for SAPUI5.

Context

You can view monitors in the following ways:

- For activated design-time objects (flowgraphs or replication tasks), in the SAP HANA Web-based Development Workbench: Editor, select the object and click the *Launch Monitoring Console* icon.
- From within a monitor, you can switch to another monitor by selecting it from *More Info* > *Navigate to (Monitor)*.
- Enter the URL address of each monitor directly into a web browser.

The following table describes the URLs for each monitor.

Monitor	How to Access
Data Provisioning Agent Monitor	`<host name>:80<2 digit instance number>/sap/hana/im/dp/monitor/index.html?view=DPAgentMonitor`
Data Provisioning Remote Source Monitor	`<host name>:80<2 digit instance number>/sap/hana/im/dp/monitor/index.html?view=DPRemoteSourceMonitor`
Data Provisioning Remote Subscription Monitor	`<host name>:80<2 digit instance number>/sap/hana/im/dp/monitor/index.html?view=DPSubscriptionMonitor`
Data Provisioning Remote Query Statistics Monitor	`<host name>:80<2 digit instance number>/sap/hana/im/dp/monitor/index.html?view=DPRemoteQueryStatisticsMonitor`

Monitor	How to Access
Data Provisioning Task Monitor	`<host name>:80<2 digit instance number>/sap/hana/im/dp/monitor/index.html?view=IMTaskMonitor`
Data Provisioning Design Time Object Monitor	`<host name>:80<2 digit instance number>/sap/hana/im/dp/monitor/index.html?view=IMDesignTimeObjectMonitor`
Data Provisioning Data Assurance Monitor	`<host name>:80<2 digit instance number>/sap/hana/im/dp/monitor/index.html?view=DPDataAssuranceMonitor`

Related Information

Grant Roles to Users

3.3 Common Concepts and Controls

The Data Provisioning monitors use several shared controls and processes.

Common controls for all monitors:

Table 12: Common Controls

Control	Description
Unit	Select to change the unit for parameters such as byte size for memory statistics or time frames for duration statistics.
Refresh	At the top, monitor level, select to refresh the data on all panes of the monitor, or you can refresh individual panes.
Auto Refresh: <x> seconds	Select the check box to enable auto refresh. Selecting *Auto Refresh* at the top, monitor level also selects the *Auto Refresh* options for all panes in the monitor. You can change the frequency for how often to refresh the pane(s). After 15 minutes, this setting automatically clears. The default auto-refresh rate for all monitor views is 30 seconds.
Clear Filter	You can apply filters on most columns or objects by selecting the column heading or object and entering filter criteria. *Clear Filter* removes the filters applied to that pane, or the *Clear Filter* button at the top, monitor level clears all filters for all panes in the monitor.

Additional common controls are available from the *More Info* menu:

Table 13: More Info Controls

Action History	Select to display a list of recent user actions such as user name, timestamp, action, and execution status of the action. These actions are stored in the table `"SAP_HANA_IM_DP"."sap.hana.im.dp.monitor.ds::DP_UI_ACTION_HISTORY"`. Users with the Operations role can truncate (clear) this table.
Documentation	Open the Data Provisioning Monitoring documentation.
History Configuration	Manage statistics logging and statistics history data retention.
Navigate to (Monitor)	Switch to another monitor by selecting it from the *Navigate to (Monitor)* menu.
Settings	General settings for the monitor such as setting a sender email address for notifications, collapsing panes, or managing user settings profiles.
System Info	Display basic information about the SAP HANA system, including SID, time zone, and so on.

For any given pane, right-click a column heading the display column view controls:

- *Sort Ascending*: Sorts the table based on the selected column
- *Sort Descending*: Sorts the table based on the selected column
- *Filter*: Filters the table based on the selected column with text you enter
- *Columns*: Select the columns to show or hide in that pane

Statistics Collection

At a regular interval, remote source and remote subscription statistics are collected by a series of procedures and stored in associated history tables:

Table 14: Logging Procedure Tables

Procedure Name	History Tables
sap.hana.im.dp.monitor.ds::LOG_MEMORY_USAGE	sap.hana.im.dp.monitor.ds::HISTORY_APPLIER_DELAY sap.hana.im.dp.monitor.ds::HISTORY_CHECK_BACKLOG sap.hana.im.dp.monitor.ds::HISTORY_CHECK_GARBAGE_COLLECTION sap.hana.im.dp.monitor.ds::HISTORY_MEMORY_USAGE sap.hana.im.dp.monitor.ds::HISTORY_SUBSCRIPTION_EXCEPTIONS sap.hana.im.dp.monitor.ds::HISTORY_SUBSCRIPTION_STATUS
sap.hana.im.dp.monitor.ds::LOG_REMOTE_QUERY_STATISTICS_FROM_VIEW	sap.hana.im.dp.monitor.ds::HISTORY_REMOTE_QUERY_STATISTICS
sap.hana.im.dp.monitor.ds::LOG_REMOTE_SOURCE_LATENCY_SP	sap.hana.im.dp.monitor.ds::HISTORY_REMOTE_SOURCE_LATENCY
sap.hana.im.dp.monitor.ds::LOG_REMOTE_SOURCE_STATISTICS	sap.hana.im.dp.monitor.ds::DP_STATISTICS
sap.hana.im.dp.monitor.ds::LOG_REMOTE_SOURCE_THROUGHPUT_SP	sap.hana.im.dp.monitor.ds::HISTORY_REMOTE_SOURCE_THROUGHPUT
sap.hana.im.dp.monitor.ds::LOG_REMOTE_SOURCE_VOLUME_SP	sap.hana.im.dp.monitor.ds::HISTORY_REMOTE_SOURCE_VOLUME

Use the *History Configuration* tool to control which types of statistics are collected:

1. Choose *More Info* > *History Configuration*.
2. In the Logging Procedures section, select a procedure to configure and click *Update*.
   The schema name, procedure name, and any other dependent procedures are displayed.
3. Choose whether to enable or disable the procedure and click *Update*.
   - *Enable*: The procedure is executed at the statistics collection interval and data is stored in the tables associated with the procedure.
   - *Disable*: The procedure is not executed at the statistics collection interval and no data is stored.

> **Note**
>
> If you disable a procedure that is required by other logging procedures, statistics data collection for the dependent procedures will also be impacted.

The remote source and remote subscription statistic *Collect Time Interval* displays on the toolbar. By default, statistics are collected every 5 minutes (300 seconds). You can change this interval to any value larger

than 60 seconds (any value less than 60 will be treated as 60 seconds) by changing the `dpserver.ini collect_interval` value as in the folowing SQL statement. Refresh your browser after changing the setting to display the new value.

```
ALTER SYSTEM ALTER CONFIGURATION ('dpserver.ini', 'SYSTEM')
SET ('remote_source_statistics','collect_interval') = '<time in seconds (default
is 300)>'
WITH RECONFIGURE;
```

Statistics History Retention and Pruning

By default, statistics data older than 15 days is automatically pruned from the history tables.

Configure a different retention period on a per-table basis with the *History Configuration* tool:

1. Choose *More Info* > *History Configuration*.
2. In the Retention Tables section, select a history table to configure and click *Update*.
3. Specify the new retention period and click *Update*.

At the next regular statistics collection interval, any history data older than the new retention period is automatically deleted from the history table with a statement such as the following:

```
DELETE FROM <table> WHERE COLLECT_TIME < ADD_DAYS (CURRENT_TIMESTAMP,
-<retention_period>);
```

For example, if you lower the retention period for a table from the default 15 days to 7 days, at the next statistics collection interval, all history data older than 7 days is deleted from the table.

Related Information

Assign Roles and Privileges
Configure User Settings Profiles [page 119]

3.4 Monitoring Agents

The Data Provisioning Agent Monitor displays system information for all agents such as status, memory details, adapter-agent mapping, and agent group information.

The *Agent Monitor* pane displays basic information such as the agent details, status, the last time it connected to the Data Provisioning server, and memory usage details.

Additional controls for this pane include:

Control	Description
Create Agent	Creates a new agent; enter the following required parameters: • Agent Name • Host name • Port
Alter Agent	Lets you modify agent parameters (for example Host, Port, Enable SSL, Agent Group) for the selected agent.
Drop Agent	Removes the selected agent from the SAP HANA system.
Add Adapters	Adds an adapter to the selected agent instance.
Update Adapters	Refreshes all adapters registered for the selected agent so that any new capabilities can be used by SAP HANA.

The following statuses are possible:

- CONNECTING
- DISCONNECTED
- CONNECTED

The *Adapter Agent Mapping* pane identifies the adapters that are associated with each agent instance. When you first open the Data Provisioning Agent Monitor, it displays information for all agents and adapters. Selecting an agent displays Adapter-Agent Mapping for that agent.

Additional controls for this pane include:

Control	Description
Remove Location	Removes an adapter from an agent instance.
Update	Updates the selected adapter for the agent so that any new capabilities can be used by SAP HANA.

The *Agent Group* pane lets you view, create, and manage agent groups.

Additional controls for this pane include:

Control	Description
Create	Creates a new agent group.
Drop	Removes the selected agent group.
Add Agents	Adds one or more agents to the selected group.

Related Information

Manage Agents from the Data Provisioning Agent Monitor [page 140]
Manage Adapters from the Data Provisioning Agent Monitor [page 142]
Create or Remove an Agent Group [page 148]
Manage Agent Nodes in an Agent Group [page 150]

3.5 Monitoring Remote Sources

The Data Provisioning Remote Source monitor lets you view each remote source's status and statistics.

Remote Source Monitor

The *Remote Source Monitor* pane displays basic information such as the adapter name, status, and subscription information. Selecting a remote source displays the statistics for that source in the following pane.

Status can be one of the following values:

- OK
- ERROR
- SUSPENDED

Statistics displayed include the following:

- *Applier Delay*: The time span of collected data that remains to be processed.
- *Scan Delay*: An estimate of the time it will take to process pending shadow table records.

Subscription information includes the following:

- Total number of subscriptions to this source
- Number of subscriptions *In progress*: Subscriptions for which the status isn't APPLY_CHANGE_DATA, CREATED or blank.
- Number of subscriptions *Replicating changes*: Subscriptions for which the status is APPLY_CHANGE_DATA.

Select a remote source and choose *Alter Remote Source* to suspend or resume the capture or distribution of the remote source.

Remote Source Statistics

The *Remote Source Statistics* pane displays details such as aggregated counts of records processed by receivers and appliers. For aggregated values, the collected time interval is 5 minutes; refer to the statistic's time stamp.

Within the *Remote Source Statistics* pane, the *Statistic Name* column contains a short description of each statistic, and a more complete description can be found in the *Comments* column. Additionally, delay-related statistics, such as *Scan delay* and *Receiver delay* are highlighted in blue.

> → Tip
>
> The *Comments* column containing a detailed description of each statistic is hidden by default.

Scan delay and *Receiver delay* are calculated as follows:

Statistic	Adapter type	Description
Scan delay	Oracle Log Reader **Note** Applies only to log reader-based replication.	Remote_Source_Current_Row_Timestamp - Remote_Source_Processed_Row_Timestamp (dpagent, adapter) Both values are in UTC.
	SAP HANA	LATEST_SCANNED_TRANS_TIME_IN_TRIG_QUEUE - LATEST_TRANS_TIME_IN_TRIG_QUEUE
Receiver delay	Oracle Log Reader **Note** Applies only to log reader-based replication.	Last received timestamp (dpserver, receiver) - Remote_Source_Processed_Row_Timestamp (dpagent, adapter)
	SAP HANA	Last received timestamp (dpserver, receiver) - LATEST_SENT_UTC_TIME_IN_APPLIER The last received timestamp value is the local server time converted to UTC.

Some statistics have corresponding graph views. Select the statistic and choose *Show Graph*. For example, say the *Data and metadata records processed by applier* statistic shows a *Statistic Value* of 83300. Selecting *Graph* shows a chart that displays the count value at each 5-minute interval over a selected time frame, such as 6 hours (*Plot values for past: 6 hours*). The summation of all these values is the *Statistic Value*.

You can select a duration (*Plot values for past:*) from 6 hours to 15 days, or change the *Graph Type*.

Graphs are available for the following statistics:

Table 15: Statistics Graphs

Statistic Type	Statistic Name
Data Provisioning Server	**Note** Graphs are available for these statistics only with SAP HANA 2.0 and newer. • Data and metadata records processed by the applier (per subscription) • Data records processed by the applier (per subscription) • INSERT records processed by the applier (per subscription) • UPSERT records processed by the applier (per subscription) • UPDATE records processed by the applier (per subscription) • DELETE records processed by the applier (per subscription) • COMMIT records processed by the applier (per subscription) • ROLLBACK records processed by the applier (per subscription) • ARCHIVE records processed by the applier (per subscription) • Data and metadata records processed by the receiver • Data and metadata records processed by the applier • Data records processed by the receiver • Data records processed by the applier

Statistic Type	Statistic Name
Data Provisioning Agent Data Provisioning Adapters	• Scan delay (seconds) • Data and metadata records processed by the receiver • Data and metadata records processed by the applier • Data records processed by the receiver • Data records processed by the applier • INSERT records processed by the applier (per subscription) • Total scans • Total trigger queue records scanned • Total scan batches • Average records per scan • Average records per batch • Total transaction • Unscanned trigger queue records • Statement memory used by the source system (MB) • Times shadow table results are retrieved • Total rowsets sent out • Total rows sent out • Current operation queue size • Total LogMiner Scan Count • Calls to standby scanner • Calls to DDL standby scanner

Monitoring End-to-End Performance

Real-time replication with SAP HANA smart data integration occurs in a series of distinct data processing steps between each component. Analyze latency, throughput, and volume for real-time replication to identify performance bottlenecks and devise mitigation strategies.

Additional Remote Source Monitor controls for monitoring end-to-end performance include:

Control	Description
Latency Ticket Management	Start, stop, or clear latency tickets for the selected remote source.
Performance Statistics	Display end-to-end performance statistics such as latency, throughput, and number of records for the selected remote source.
Volume of Records	Display the history of record volume processed by the applier for a remote source.

Related Information

Measuring Latency [page 32]

Measuring Throughput [page 34]
Measuring Record Volume [page 36]
Common Concepts and Controls [page 23]

3.5.1 Measuring Latency

Monitor end-to-end latency between the source database and the target SAP HANA system with latency tickets created by the LATENCY MONITORING remote source capability.

Latency statistics for scanner, sender, receiver, and applier are available for the following adapters:

- ASELTLAdapter
- HanaAdapter
- DB2LogReader (log-based)
- DB2Mainframe
- MssqlLogReader
- OracleLogReader

> ⚠ Restriction
>
> Latency monitoring tracks only the general latency between the agent and the source database. There is currently no capability for tracking specific rows from source to target.

When latency monitoring is enabled, a ticket is written to the source database and then detected and processed by the adapter. Information about how and when each component processes the ticket is stored in the M_REMOTE_SOURCE_LATENCY_HISTORY view in the target SAP HANA database.

Each of the following steps is timestamped for tracking:

1. At fixed intervals, SAP HANA sends a request to the adapter to create a latency ticket.
2. Adapter creates and records the ticket in the source database.
3. Adapter reads the ticket and sends it back to the server.
4. Server records that it received the returned ticket and each server subcomponent timestamps the ticket until it reaches the applier.
   The end-to-end agent latency is the difference between the timestamps for when the ticket is written to the database and when the ticket is executed at the primary database.

Managing Latency Tickets

In the Remote Source Monitor, select the remote source and click *Latency Ticket Management* to open the Latency Ticket Status view.

Action	Description
Create ticket	In the Latency Ticket Status view, click *Create*. Specify the name and interval for the ticket, and click *Start*. → Tip If you do not specify an interval, a ticket with a one-time execution is created. The latency ticket is created with the following SQL command and appears in the Latency Ticket Status view: `ALTER REMOTE SOURCE <remote_source_name> START LATENCY MONITORING '<ticket_name>' [INTERVAL <interval_in_seconds>];`
Stop ticket	In the Latency Ticket Status view, select the ticket and click *Stop*. The latency ticket is stopped with the following SQL command: `ALTER REMOTE SOURCE <remote_source_name> STOP LATENCY MONITORING '<ticket_name>';`
Clear history	In the Latency Ticket Status view, select the ticket and click *Clear*. The latency ticket history is cleared with the following SQL command: `ALTER REMOTE SOURCE <remote_source_name> CLEAR LATENCY HISTORY '<ticket_name>';` Latency history data is removed from the `M_REMOTE_SOURCE_LATENCY_HISTORY` view.

Automatic Latency Ticket Creation

At 5 minute intervals, a latency ticket named “DP_TOKEN” with an interval of 30 minutes is created for each remote source with at least one active remote subscription, if it does not already exist.

Monitoring Latency

In the Remote Source Monitor, select the remote source and click *Performance Statistics* to open the End-to-End Performance Measurement view. Latency statistics calculated based on operation timestamps stored in the latency ticket history appear in the Latency section.

Observe latency trends rather than looking at momentary snapshots. For example, consider whether the agent is falling behind or catching up. If the agent is catching up and latency is decreasing, performance may be sufficient and no additional tuning may be needed. If it is falling behind and latency is increasing, consider tuning your remote source.

Latency History

At the statistics collection interval (default: 5 minutes), latency statistics for scanner, sender, receiver, and applier are calculated and stored in the `"SAP_HANA_IM_DP"."sap.hana.im.dp.monitor.ds::HISTORY_REMOTE_SOURCE_LATENCY"` table.

Table 16: HISTORY_REMOTE_SOURCE_LATENCY Columns

Column	Description
REMOTE_SOURCE_NAME	Remote source name
ADAPTER_NAME	Adapter name
LATENCY_TICKET_NAME	Defined manually when ticket monitoring is started
TICKET_TIME	Timestamp the ticket was created
T0_DB_TRANSACTION_TIME	Timestamp the transaction occurred in the source DB
T1_AGENT_SCAN_TIME	Timestamp the adapter scanned the row successfully
T2_AGENT_SEND_TIME	Timestamp the adapter sent the row to the HANA dpserver
T3_SERVER_RECEIVE_TIME	Timestamp the dpserver received the row successfully
T4_SERVER_APPLY_TIME	Timestamp the dpserver committed the row to the target table
L1_AGENT_SCAN_MSEC	Milliseconds between T0 and T1
L2_AGENT_QUEUE_MSEC	Milliseconds between T1 and T2
L3_RECEIVER_MSEC	Milliseconds between T2 and T3
L4_APPLIER_MSEC	Milliseconds between T3 and T4
E2E_LATENCY_MSEC	Milliseconds between T0 and T4

> **Note**
>
> `TICKET_TIME` is recorded in the local server time. All other timestamp fields are recorded in UTC.

> **Restriction**
>
> For the DB2Mainframe adapter, scanner and sender latency cannot be calculated separately. In entries for this adapter, scanner is empty, and the value shown for sender is the combined latency for scanner and sender.

3.5.2 Measuring Throughput

Analyze throughput, a measure of communication or process efficiency, to expose bottlenecks and determine whether high latency may be justified by high processing volume.

Throughput statistics for scanner, sender, receiver, and applier are available for the following adapters:

- HanaAdapter
- OracleLogReader

- MssqlLogReader
- DB2LogReader (log-based)
- DB2Mainframe

Monitoring Throughput

In the Remote Source Monitor, select the remote source and click *Performance Statistics* to open the End-to-End Performance Measurement view. Statistics stored in the throughput history appear in the Throughput section.

Throughput values are calculated and displayed in rows per millisecond (rows/ms).

Throughput History

At the statistics collection interval (default: 5 minutes), throughput statistics for scanner, sender, receiver, and applier are calculated and stored in the `"SAP_HANA_IM_DP"."sap.hana.im.dp.monitor.ds::HISTORY_REMOTE_SOURCE_THROUGHPUT"` table.

Table 17: HISTORY_REMOTE_SOURCE_THROUGHPUT Columns

Column	Description
REMOTE_SOURCE_NAME	Remote source name
ADAPTER_NAME	Adapter name
COLLECT_TIME	Timestamp the statistic was recorded
T1_AGENT_SCAN	Scanner throughtput in rows/ms
T2_AGENT_QUEUE	Sender throughput in rows/ms
T3_RECEIVER	Receiver throughput in rows/ms
T4_APPLIER	Applier throughput in rows/ms

> ⓘ Note
>
> `COLLECT_TIME` is recorded in the local server time.

> ⚠ Restriction
>
> Receiver and applier throughputs are available only with SAP HANA revision 2.00.045 and newer.

> ⚠ Restriction
>
> Applier throughput is available only with the following SAP HANA revisions:
>
> - 2.00.048.01 and newer
> - 2.00.052 and newer

3.5.3 Measuring Record Volume

Analyze record volume to determine the number of records processed by each component.

Volume statistics for scanner, sender, receiver, and applier are available for the following adapters:

- HanaAdapter
- OracleLogReader
- MssqlLogReader
- DB2LogReader (log-based)
- DB2Mainframe

Monitoring Record Volume

In the Remote Source Monitor, select the remote source and click *Performance Statistics* to open the End-to-End Performance Measurement view. Statistics stored in the volume history appear in the Number of Rows section.

Record Volume History

At the statistics collection interval (default: 5 minutes), record volume statistics are calculated and stored in the `"SAP_HANA_IM_DP"."sap.hana.im.dp.monitor.ds::HISTORY_REMOTE_SOURCE_VOLUME"` table.

Table 18: HISTORY_REMOTE_SOURCE_VOLUME Columns

Column	Description
REMOTE_SOURCE_NAME	Remote source name
ADAPTER_NAME	Adapter name
COLLECT_TIME	Timestamp the statistic was recorded
V1_AGENT_SCAN_ROWCOUNT	Volume scanned
V2_AGENT_QUEUE_ROW-COUNT	Volume sent
V3_RECEIVER_ROWCOUNT	Volume received
V4_APPLIER_ROWCOUNT	Volume applied

> **Note**
>
> `COLLECT_TIME` is recorded in the local server time. All volumes are counted over the latest collection interval.

> **Note**
>
> For the Oracle, Microsoft SQL Server, and DB2 log reader adapters in log reader mode, the volume scanned and volume sent statistics include both data and metadata records. In trigger-based mode, and for the SAP HANA adapter, these statistics include only data records.

> **Restriction**
>
> - Receiver and applier volumes are available only with SAP HANA revision 2.00.045 and newer.

Identify High-Volume Remote Sources and Subscriptions

The processing speed for a single remote source can be reduced by UPDATE and DELETE operations because these operations are applied one row at a time and cannot be batched. If you have a very large volume of records, consider splitting the subscriptions under a remote source into multiple remote sources to take advantage of parallel processing.

In the Remote Source Monitor, click *Volume of Records Applied* to view detailed volume information:

- Per remote source
  Identify remote sources that are processing large numbers of records over the specified date range.
- Per remote subscription
  Identify remote subscriptions that are processing large numbers of records over the specified date range. Consider splitting these subscriptions over multiple remote sources.

3.6 Monitoring Remote Subscriptions

The Data Provisioning Remote Subscription Monitor provides information about how data is being replicated to the Data Provisioning Server.

Remote Subscription Monitor

The *Remote Subscription Monitor* pane displays basic information such as the design-time information, status, and subscription type. You can select a subscription name to view its description. Selecting a subscription displays the statistics for that subscription in the pane below.

Additional controls for this pane include:

Control	Description
Queue	Initiates real-time data processing
Distribute	Applies changes

Control	Description
Reset	Resets the real-time process to start from the initial load again
Drop	Removes the selected remote subscription.
Notifications	Select to view or add email notifications (for example in the case of a warning or error).

Remote Subscription Statistics

Within the *Remote Subscription Statistics* pane, the *Statistic Name* column contains a short description of each statistic, and a more complete description can be found in the *Comments* column. Additionally, delay-related statistics, such as *Applier delay* and *Receiver delay* are highlighted in blue.

Some statistics have corresponding graph views. Select the statistic and choose *Show Graph*. Selecting *Graph* shows a chart with the count value at each 5-minute interval over a selected time frame, for example 6 hours (*Plot values for past: 6 hours*).The summation of all these values is the *Statistic Value*.

You can select a duration (*Plot values for past:*) from 6 hours to 15 days, or change the *Graph Type*.

Graphs are available for the following statistics:

Table 19: Statistics Graphs

Statistic Type	Statistic Name
Data Provisioning Server	**Note** Graphs are available for these statistics only with SAP HANA 2.0 and newer. • Data and metadata records processed by applier (per subscription) • Data records processed by applier (per subscription) • INSERT records processed by applier (per subscription) • UPSERT records processed by applier (per subscription) • UPDATE records processed by applier (per subscription) • DELETE records processed by applier (per subscription) • COMMIT records processed by applier (per subscription) • ROLLBACK records processed by applier (per subscription) • ARCHIVE records processed by applier (per subscription)

Statistic Type	Statistic Name
Data Provisioning Agent Data Provisioning Adapters	• Records in shadow table • UPDATE records in source • INSERT records in source • DELETE records in source

Related Information

Remote Subscription Statuses [page 39]
Processing Remote Source or Remote Subscription Exceptions [page 165]
Suspend and Resume Remote Sources [page 162]
Manage Remote Subscriptions [page 164]
Common Concepts and Controls [page 23]
Monitoring Remote Sources [page 28]
Manage Remote Subscriptions [page 164]
Create Notifications [page 120]

3.6.1 Remote Subscription Statuses

You create remote subscriptions when you want to listen for real-time changes to data that you replicated into the Data Provisioning Server. A remote subscription can have seven different statuses.

The following table describes the status names (in both the *Data Provisioning Remote Subscription Monitor* and in the M_REMOTE_SUBSCRIPTIONS monitoring view) and their descriptions.

You can select a status entry to display the *Replication Status Details* window for information about queueing and distribution statistics:

- Queue command: Adapter received begin marker from the Data Provisioning Server
- Queue command: Adapter applied begin marker to source
- Queue command: Adapter read begin marker from source and sent to the Data Provisioning Server
- Distribute command: Adapter received end marker from the Data Provisioning Server
- Distribute command: Adapter applied end marker to source
- Distribute command: Adapter read end marker from source and sent to the Data Provisioning Server

The number of transactions collected during initial load and still pending application in auto-correct mode is displayed at the top of the *Replication Status Details* window.

Remote Subscription Monitor	M_REMOTE_SUBSCRIPTIONS	Description
Created	CREATED	Remote subscription is created by the replication task or flowgraph.

Remote Subscription Monitor	M_REMOTE_SUBSCRIPTIONS	Description
Subscribed, request to queue	MAT_START_BEG_MARKER	The receiver is waiting for the begin marker that indicates the first changed data to queue while the initial load is running.
Request to stop queuing and start distribution	MAT_START_END_MARKER	The receiver queues the rows and is waiting for the end marker that indicates the last row of the initial load.
Queuing changes	MAT_COMP_BEG_MARKER	The receiver has received the begin marker, change data is being queued, and the initial load can now start.
Queuing closed, starting distribution	MAT_COMP_END_MARKER	The receiver queues the changed rows and is waiting for the end marker that indicates the last row of the initial load. The initial load has completed and the end marker is sent to the adapter. If the state doesn't change to AUTO_CORRECT_CHANGE_DATA, the adapter or source system is slow in capturing the changes.
Applying queued changes using auto-correct	AUTO_CORRECT_CHANGE_DATA	When the end marker is received, the applier loads the changed data captured (and queued during the initial load) to the target. If many changes occurred after the initial load started, this state might take a long time to change to APPLY_CHANGE_DATA.
Replicating changes	APPLY_CHANGE_DATA	All of the changes captured while the initial load was running have completed and are now loaded to the target. The subscription is now applying changes as they happen in the source system.

3.7 Monitoring Remote Queries

The Data Provisioning Remote Query Statistics monitor provides information about the remote query statements processed for each remote source.

Prerequisites

To populate the Data Provisioning Remote Query Statistic, you must complete the following prerequisite tasks for each Data Provisioning Agent that performs initial loads:

1. Add the AgentMgmtAdapter to each agent instance.
2. Create a remote source for the AgentMgmtAdapter.
3. Create three virtual tables with a postfix that is unique to each agent instance.
   - `QuerySqlStatements --> "SAP_HANA_IM_DP"."VT_AGENT_SQL_STATISTICS<postfix>"`
   - `QueryMemoryStatistics --> "SAP_HANA_IM_DP"."VT_AGENT_MEMORY_STATISTICS<postfix>"`

- `QueryExecutionStatistics --> "SAP_HANA_IM_DP"."VT_AGENT_QUERY_STATISTICS<postfix>"`

For example, for three agent instances, you must create a total of nine virtual tables.

> → Remember
>
> Each virtual table must be created in the **SAP_HANA_IM_DP** schema, and must match the naming above.

Remote Query Overview

The *Remote Query Overview* pane displays basic information such as the SAP HANA statement, remote statement executed on the source, and basic statistical information such as the start and end time, and aggregate throughput. Queries are displayed in pages, with the most recent queries displayed first.

Select a remote query to display additional details in the *Remote Query Statistics* pane below.

Additional controls for this pane include:

Control	Description
Show Graph	Display a graph view for a statistic for the selected query with the value at each 5-minute interval over a selected time frame, for example 6 hours (*Plot values for past: 6 hours*). You can select a duration (*Plot values for past:*) from 6 hours to 15 days, change the *Graph Type*, and choose which statistic to graph.
Prev	Returns to the previous page of remote queries
Next	Advances to the next page of remote queries

Table 20: Remote Query Overview Columns

Column Name	Description
Statement ID	Statement ID
Remote Source Schema Name	Schema name for the remote source
Remote Source Name	Name of the remote source
Agent Name	Name of the data provisioning agent instance

Column Name	Description
Status	Statement status, highlighted green, yellow, or red: • STARTING • EXECUTING • FETCHING • CANCELLING • CANCELLED • ERROR • CLOSED For queries with an ERROR status, you can click the status to view the associated error message.
Start Time	Time the statement started
End Time	Time the statement was closed
Received Record Count	Number of records received by the data provisioning server
Fetch Size	Number of records to fetch at a time
Statement Runtime	Cumulative statement runtime `End Time - Start Time`
Network Throughput	Network throughput in bytes/ms `Received Size / Result Set Send Duration`
Query Throughput	End-to-end query throughput in rows/ms `Records Received from Agent / Statement Run Time`
Indexserver Throughput	Index server throughput in bytes/ms `Received Size / (Total Response Count * Average Request Idle Duration)`
Virtual Table Names	Relevant virtual table names accessed by the query Comma-separated list in the format: `<schema>.<name>`
Statement	SAP HANA statement strinng
Remote Statement	Rewritten remote statement string

Remote Query Statistics

The *Remote Query Statistics* pane displays additional detailed statistical information for the remote query on both the *Data Provisioning Server* and *Data Provisioning Agent*.

Table 21: Data Provisioning Server

Column Name	Description
User Name	User Name
Received Size	Size of records received by the Data Provisioning Server
Delivered Record Count	Number of records delivered to the Index Server
Delivered Size	Size of records delivered to the Index Server
Average Queue Size	Average memory used by the Data Provisioning Server prefetch queue
Peak Queue Size	Peak memory used by the Data Provisioning Server prefetch queue
Peak Time	Time when the prefetch queue reached peak usage
Average Request Idle Duration	Average time spent waiting for fetch requests from the Index Server
Average Request Servicing Duration	Average time spent servicing fetch requests from the Index Server

Table 22: Data Provisioning Agent

Column Name	Description
Is Streaming	Flag indicating results are streamed
Sent Record Count	Number of records sent by the Data Provisioning Agent
Max Record Size	Maximum potential size of a record
Start Time	Agent statement start time
End Time	Time when the statement is closed on the agent
Rewrite Duration	Time taken to rewrite the statement
Execute Duration	Time taken to execute the statement
Fetch Duration	Time taken to fetch results
Serialize Duration	Time taken to serialize results
Send Duration	Time taken to send results
Serializer Peak Count	Peak number of items in the serializer queue
Serializer Peak Size	Peak memory used by the serializer queue
Serializer Peak Time	Time when the serializer queue reached peak usage
Serializer Total Count	Total count of all objects passed through the serializer queue
Serializer Total Size	Total size of all objects passed through the serializer queue
Serializer Pool Capacity	Capacity of the serializer shared memory pool
Serializer Pool Peak Size	Peak memory used by the serializer shared memory pool during statement execution
Serializer Pool Peak Time	Time when the serializer pool reached peak usage
Response Peak Count	Number of objects in the response queue at peak usage time
Response Peak Size	Peak memory used by the response queue
Response Peak Time	Time when the response queue reached peak usage
Repsonse Total Count	Total count of all objects passed through the response queue

Column Name	Description
Response Total Size	Total size of all objects passed through the response queue
Response Pool Capacity	Capacity of the response shared memory pool
Response Pool Peak Size	Peak memory used by the response shared memory pool during statement execution
Repsonse Pool Peak Time	Time when the response pool reached peak usage

3.8 Monitoring Design-Time Objects

The Data Provisioning Design Time Object Monitor provides information about your design-time objects (flowgraphs and replication tasks). For example, you can see the duration of a task execution for a flowgraph and how many records have been processed.

The *Design Time Objects* pane displays basic information such as the target schema name and whether the object has a table type input or variables. Selecting a design-time object displays task and remote subscription information for that object in the panes below.

Additional controls for this pane include:

Control	Description
Execute	Executes the selected object.
Schedules	Manage scheduling for the selected object.
Notifications	Select to view or add email notifications (for example in the case of a warning or error).

The *Task Monitor* pane displays task duration and number of processed records.

Additional controls for this pane include:

Control	Description
Stop	Stop execution of the selected task.
Remote Statements	For a selected task, if the *Remote Statements* button is enabled, select it to view the SQL remote statement string used for task execution.
Notifications	Select to view or add email notifications (for example in the case of a warning or error).

Selecting a task displays the *Data Provisioning Task Monitor*.

The *Remote Subscription Monitor* pane displays subscription status and other processing details. Select a subscription name to launch the *Data Provisioning Remote Subscription Monitor* Select a status entry to display a *Details* dialog for more details including marker information and exceptions.

Related Information

Execute Flowgraphs and Replication Tasks [page 166]

Schedule Flowgraphs and Replication Tasks [page 167]
Stop Non-Realtime Flowgraph Executions [page 169]
Execute Flowgraphs and Replication Tasks [page 166]
Create Notifications [page 120]
Monitoring Tasks [page 45]
Monitoring Remote Subscriptions [page 37]
Managing Design Time Objects [page 165]

3.9 Monitoring Tasks

The *Data Provisioning Task Monitor* provides you with information about your replication (hdbreptask) and transformation (hdbflowgraph) tasks. For example, you can see the duration of a task execution and how many records have been processed.

When you first open the *Data Provisioning Task Monitor*, it displays information for all tasks.

The *Task Overview* pane lists all tasks with information such as design time name (select to open the Design-Time Objects monitor), create time, and memory size, for example. You can select a task to show information for only that task in the panes below.

Additional controls for this pane include:

Control	Description
Start	Start execution of the selected task.
Schedules	Manage scheduling for the selected object.
Notifications	Select to view or add email notifications (for example in the case of a warning or error).

The *Task Execution Monitor* pane lists all of the recent executions of the task(s) and associated data such as the schema name, start time, duration, status, and number of processed records.

Additional controls for this pane include:

Control	Description
Stop	Stop execution of the selected task.
Remote Statements	For a selected task, if the *Remote Statements* button is enabled, select it to view the SQL remote statement string used for task execution.
Execute Remaining Partitions	Execute partitions that didn't run or failed after executing a task.

The *Task Operation Execution Monitor* pane lists all of the recent operations of the task(s) and associated data including the operation type, start time, duration, status, and number of processed records.

Related Information

Change Retention Period for Data Provisioning Task Monitor [page 46]

Start and Stop Data Provisioning Tasks [page 170]
Schedule Data Provisioning Tasks [page 171]
Executing Partitions [page 172]
Start and Stop Data Provisioning Tasks [page 170]
Schedule Flowgraphs and Replication Tasks [page 167]

3.9.1 Change Retention Period for Data Provisioning Task Monitor

Change the retention period for task statistics tables if you want to retain the data in the Task Execution Monitor and Task Operation Execution Monitor for longer than 90 days.

Context

The following parameters specify how long to keep the statistics data and when to delete them:

- The `task_data_retention_period` parameter specifies the period of time the data remains in the statistics tables, in seconds. This period is calculated from the time the task reached the COMPLETED, FAILED, or CANCELLED status. The default value is 7776000, or 90 days. A value of 0 (zero) or -1 means never delete the data.
- The `task_data_retention_period_check_interval` parameter specifies how often the data is deleted by the garbage collection thread. The default value is 300 seconds (5 minutes).

To change the default values of these parameters, you must add a new section named task_framework to each of the `indexserver.ini`, `scriptserver.ini`, and the `xsengine.ini` files.

To change these options in the SQL console:

```
ALTER SYSTEM ALTER CONFIGURATION ('<server type>.ini', 'SYSTEM')
SET ('task_framework', 'task_data_retention_period_check_interval')='<in secs>'
WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('<server type>.ini', 'SYSTEM')
SET ('task_framework', 'task_data_retention_period')='<in secs>'
WITH RECONFIGURE;
```

Procedure

1. Log in to SAP HANA studio as SYSTEM user.
2. In the *Systems* view, right-click the name of your SAP HANA server and choose *Configuration and Monitoring* > *Open Administration*.
3. Click the *Configuration* tab.
4. Right-click *indexserver.ini* and choose *Add Section*.

5. On the *Section Name* screen of the *Add Section Wizard*, enter **`task_framework`** for the section name, and click *Next*.
6. On the *Scope Selection* screen, select *System* from the *Assign Values to*dropdown list, and click *Next*.
7. On the *Key Value Pairs* screen, enter **`task_data_retention_period`** in the *Key* field, and enter the number of seconds you want the statistics data to be kept in the statistics tables.

   > **Note**
   >
   > If you set the retention period to 0 (zero) or -1, statistics data is not deleted.

8. Click *Finish*.
9. Select *indexserver.ini*, right-click, and choose *Add Parameter*.
10. On the *Add New Parameter* screen, enter **`task_data_retention_period_check_interval`** in the *Key* field, and enter the time interval for the Task Data Cleanup Process to run.

    > **Note**
    >
    If you set the `task_data_retention_period_check_interval` to less than 60 seconds, the default value (5 minutes) is used.

11. Repeat steps 4 through 10 for *scriptserver.ini*.
12. If your tasks are called by the XS application, select *xsengine.ini* and repeat steps 4 through 10.

3.10 Monitoring Statistics Reference

Monitoring statistics provide insight into the status and performance of your SAP HANA smart data integration components, objects, and operations.

Table 23: Available Monitoring Statistics

ID	Language	Statistic Name	Unit	Long Description	Component
00001	EN	Applier delay (s)	seconds	Time taken for a row received from DPAgent to be applied by DPServer to the target (table or stored procedure), in seconds. Applier delay is calculated at the remote source level by comparing the "last received" timestamp and the "last applied" timestamp. Therefore, the applier delay represents the time span of data that remains to be processed, rather than the amount of time that it will take to process change data sent by the agent.	
00002	EN	Receiver delay (s)	seconds	Time taken for a row sent from DPAgent to be received by DPServer, in seconds. This time indicates the delay between receiving the changed data record from the DPFramework and placing it into the receiver queue.	

ID	Language	Statistic Name	Unit	Long Description	Component
00003	EN	Scan delay (s)	seconds	Time taken for a row from the source database to be scanned by DPAgent, in seconds. This time indicates the delay in reading records by DPAgent from the source. For trigger-based adapters, scan delay calculations are based on averages of values collected at a given time. Therefore, the delay represents a best estimate for the time it will take to process pending shadow table records. For each adapter, the formula used to calculate scan delay is the same, but the individual statistics vary. `SCAN_DELAY = (AVG_TIME_TO_SCAN_TRIGGER_QUEUE / AVG_RECORDS_PER_SCAN + AVG_SCAN_TIME_PER_RECORD) * NUMBER_OF_UNSCANNED_RECORDS` • SAP HANA adapter: `SCAN_DELAY = (30003/30`	

ID	Language	Statistic Name	Unit	Long Description	Component
				`006 + 30008) * 30010` • Oracle adapter: `SCAN_DELAY = (40113/40116+40118) * 40120` • Microsoft SQL Server adapter: `SCAN_DELAY = (50086/50089 + 50091) * 50093` • IBM DB2 adapter: `SCAN_DELAY = (60042/60045 + 60047) * 60049` • IBM DB2 Mainframe adapter: `SCAN DELAY = (70051/70054 + 70056) * 70058`	
10000	EN	There are no active remote subscriptions on the remote source	n/a	There are no active remote subscriptions on the remote source.	

ID	Language	Statistic Name	Unit	Long Description	Component
10001	EN	Average receiver record processing time (microseconds)	microseconds	Average time taken for a record to be received from DPFramework and placed into the receiver's queue, in microseconds.	
10002	EN	Average distributor record processing time (microseconds)	microseconds	Average time taken for a record to be read from the receiver's queue, in microseconds. These transactions have been committed on the source and need to be applied on the target.	
10003	EN	Average applier record processing time (microseconds)	microseconds	Average time taken for a record and respective DML statement (INSERT, UPDATE, or DELETE) to be applied from receiver to the target (table or stored procedure), in microseconds.	
10004	EN	Average applier record processing time per subscription (microseconds)	microseconds	Average time taken for a record and respective DML statement (INSERT, UPDATE, or DELETE) to be applied from receiver to the target (table or stored procedure), per subscription in microseconds.	
10006	EN	Data and metadata records processed by receiver	count	Number of all data and metadata records received from DPFramework and placed into receiver's queue.	

ID	Language	Statistic Name	Unit	Long Description	Component
10007	EN	Data and metadata records processed by distributor	count	Number of all data and metadata records read from the receiver's queue. These transactions have been committed on the source and need to be applied on the target.	
10008	EN	Data and metadata records processed by applier	count	Number of all data and metadata records and respective DML statements (INSERT, UPDATE, or DELETE) applied from receiver to the target (table or stored procedure).	
10009	EN	Data and metadata records processed by applier (per subscription)	count	Number of all data and metadata records and respective DML statements (INSERT, UPDATE, or DELETE) applied from receiver to the target (table or stored procedure), per subscription.	
10011	EN	Data records processed by receiver	count	Number of data records received from DPFramework and placed into the receiver's queue.	
10012	EN	Data records processed by distributor	count	Number of data records read from the receiver's queue. These transactions have been committed on the source and need to be applied on the target.	

ID	Language	Statistic Name	Unit	Long Description	Component
10013	EN	Data records processed by applier	count	Number of data records and respective DML statements (INSERT, UPDATE, or DELETE) applied from receiver to the target (table or stored procedure).	
10014	EN	Data records processed by applier (per subscription)	count	Number of data records and respective DML statements (INSERT, UPDATE, or DELETE) applied from receiver to the target (table or stored procedure), per subscription.	
10016	EN	Last received timestamp	timestamp	Timestamp assigned to the record when received from the agent and placed into receiver's queue.	
10017	EN	Last distributed timestamp	timestamp	Timestamp of the record (DML) that was last read by the distributor component from the receiver component's queue. **Note** The timestamp is assigned to the record as soon as the receiver component receives the record from the agent.	

ID	Language	Statistic Name	Unit	Long Description	Component
10018	EN	Last applied timestamp	timestamp	Timestamp of the record (DML) that was last applied from receiver to the target (table or stored procedure). **Note** The timestamp is assigned to the record as soon as receiver component receives the record from agent.	
10019	EN	Last applied timestamp (per subscription)	timestamp	(Per subscription) Timestamp of the record (DML) that was last applied from receiver to the target (table or stored procedure). **Note** The timestamp is assigned to the record as soon as receiver component receives the record from agent.	
10021	EN	INSERT records processed by applier (per subscription)	count	Number of INSERT records applied from receiver to the target (table or stored procedure), per subscription.	
10022	EN	UPSERT records processed by applier (per subscription)	count	Number of UPSERT applied from receiver to the target (table or stored procedure), per subscription.	

ID	Language	Statistic Name	Unit	Long Description	Component
10023	EN	UPDATE records processed by applier (per subscription)	count	Number of UPDATE records applied from receiver to the target (table or stored procedure), per subscription.	
10024	EN	DELETE records processed by applier (per subscription)	count	Number of DELETE records applied from receiver to the target (table or stored procedure), per subscription.	
10025	EN	COMMIT records processed by applier (per subscription)	count	Number of COMMIT records applied from receiver to the target (table or stored procedure), per subscription.	
10026	EN	ROLLBACK records processed by applier (per subscription)	count	Number of ROLLBACK records applied from receiver to the target (table or stored procedure), per subscription.	
10027	EN	Last updated time	timestamp	Timestamp when the adapter provided updated statistics.	
10029	EN	ARCHIVE records processed by applier (per subscription)	count	Number of ARCHIVE records applied from receiver to the target (table or stored procedure), per subscription.	
30001	EN	Total scans	count	Cumulative number of total scans, starting from the most recent start or resumption of the adapter instance. This number increases if any of the remote source tables have continuous changes.	HanaAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
30002	EN	Invalid scans	count	Cumulative number of invalid scans, starting from the most recent start or resumption of the adapter instance. If this number is increasing, there are no changes to any of the source tables of a remote source.	HanaAdapter
30003	EN	Average time to scan trigger queue (ms)	milliseconds	Average time taken to scan the trigger queue, starting from the most recent start or resumption of the adapter instance, in milliseconds. This value is calculated for the active processing time, and doesn't include time when the adapter instance was idle. This average is calculated by dividing the total cumulative scan time by the number of total scans.	HanaAdapter
30004	EN	Total trigger queue records scanned	count	Total number of records scanned by the adapter for a remote source.	HanaAdapter
30005	EN	Total scan batches	count	In each scan, trigger queue records are split into batches based on table name and max batch size.	HanaAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
30006	EN	Average records per scan	count	If the number of unscanned trigger queue records is large and the average records per scan is much smaller than the maximum scan size configured in the remote source, there are too many small transactions per scan. In this case, increase the maximum transaction count per scan in the remote source configuration.	HanaAdapter
30007	EN	Average records per batch	count	Average number of records per batch. This average is calculated by dividing the total number of trigger queue records scanned by the total number of scan batches.	HanaAdapter
30008	EN	Average time to scan trigger queue, per record (ms)	milliseconds	Average time taken per record to retrieve the trigger queue and split into different batches, in milliseconds.	HanaAdapter
30009	EN	Total transactions	count	Total number of transactions processed.	HanaAdapter
30010	EN	Unscanned trigger queue records	count	Current number of unscanned trigger queue records. If this value is decreasing, replication is catching up with the source changes. Otherwise, replication may lag.	HanaAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
30011	EN	Latest scanned trans_time in trigger queue	timestamp	Latest scanned trans_time in the trigger queue. If this value is close to 'Latest trans_time in trigger queue', there is no lag in scanning.	HanaAdapter
30012	EN	Latest trans_time in trigger queue	timestamp	Latest trans_time (that is, max) in the trigger queue. If this value changes continuously, there are changes to source tables in this remote source.	HanaAdapter
30013	EN	Max scan number in trigger queue	number	Max scan number in trigger queue.	HanaAdapter
30014	EN	Average transaction size	count	Overall average transaction size since the remote source started.	HanaAdapter
30015	EN	Dispersion of transaction size	count	Standard variance of transaction size since the remote source started.	HanaAdapter
30017	EN	Statement memory used by the source system (MB)	megabytes	Memory consumption in the source system for the statements executed by HanaAdapter, in megabytes. To enable statement memory tracking, set the following parameters to **on** in the source system's `global.ini` file: • `enable_tracking` • `memory tracking`	HanaAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
30018	EN	Times shadow table results are retrieved	count	Number of times the shadow table results have been retrieved.	HanaAdapter
30019	EN	Average time to retrieve shadow table results (ms)	milliseconds	If this value is increasing over time, it indicates that the source database may be slow to respond.	HanaAdapter
30020	EN	Total rowsets sent out	count	A row set can have multiple rows up to max batch size	HanaAdapter
30021	EN	Total rows sent out	count	Total count of rows sent out	HanaAdapter
30022	EN	Average rows per rowset	count	Average number of rows per rowset, calculated by dividing the total count of rows sent out by the total count of rowsets sent out. When there are many small transactions, this value is closer to 1.0.	HanaAdapter
30023	EN	Average time to send rowsets (ms)	milliseconds	Average time to send rowsets to DPServer, in milliseconds.	HanaAdapter
30024	EN	Latest timestamp applier sent rows	timestamp	Most recent time the adapter's applier sent rows to DPServer.	HanaAdapter
30025	EN	Latest timestamp applier sent rows (UTC)	timestamp	Most recent time the adapter's applier sent rows to DPServer (UTC).	HanaAdapter
30026	EN	Latest applier sent sequence ID	timestamp	Sequence ID of the last row the adapter's applier has sent to DPServer.	HanaAdapter
30027	EN	Latest received truncated sequence ID	timestamp	Sequence ID of the last row received by the DPServer.	HanaAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
30028	EN	Begin marker received	timestamp	Timestamp when the adapter received the begin marker request from DPServer.	HanaAdapter
30029	EN	Begin marker applied	timestamp	Timestamp when the begin marker was applied to the source.	HanaAdapter
30030	EN	Begin marker sent	timestamp	Timestamp when the begin marker was read from the source and sent to DPServer.	HanaAdapter
30031	EN	End marker received	timestamp	Timestamp when the end marker was received by the adapter from DPServer.	HanaAdapter
30032	EN	End marker applied	timestamp	Timestamp when the end marker was applied to the source.	HanaAdapter
30033	EN	End marker sent	timestamp	Timestamp when the end marker was read from the source and sent to DPServer.	HanaAdapter
30037	EN	Max scan time per record over the latest statistics update interval (ms)	milliseconds	If this value is larger than 0 seconds, the source database may be busy.	HanaAdapter
30038	EN	Trigger queue records in the slowest scan	count	Number of trigger queue records in the slowest scan over the latest statistics update interval (MAX_SCAN_TIME_PER_RECORD equals the value above).	HanaAdapter
30039	EN	Min scan time per record over latest statistics update interval (ms)	milliseconds	If this value is larger than 0 seconds, the source database may be busy.	HanaAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
30040	EN	Trigger queue records in the fastest scan	count	Number of trigger queue records in the fastest scan over the latest statistics update interval (MIN_SCAN_TIME_PER_RECORD equals the value above).	HanaAdapter
30041	EN	Records in shadow table	count	Value string: <number of records in shadow table>;<remotetableID>;"<shadow table schema>"."<shadow table name>"	HanaAdapter
30042	EN	UPDATE records in source	count	Value string: <number of UPDATE records in source>; <remotetableID>;"<source table schema>"."<source table name>" If the source database is restarted, this value is reset.	HanaAdapter
30043	EN	INSERT records in source	count	Value string: <number of INSERT records in source>; <remotetableID>;"<source table schema>"."<source table name>" If the source database is restarted, this value is reset.	HanaAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
30044	EN	DELETE records in source	count	Value string: <number of DELETE records in source>; <remote-tableID>;"<source table schema>". "<source table name>" If the source database is restarted, this value is reset.	HanaAdapter
40001	EN	Time statistics obtained	timestamp	Day, date, and time when statistics were obtained.	OracleLogReader-Adapter
40002	EN	Time replication last started	timestamp	Day, date, and time that the **REPLICATING** state started.	OracleLogReader-Adapter
40003	EN	Time statistics last reset	timestamp	Day, date, and time that statistics counters were reset.	OracleLogReader-Adapter
40004	EN	Items held in Global LRUCache	count	Number of object references in the internal Least Recently Used cache.	OracleLogReader-Adapter
40005	EN	VM maximum memory (B)	bytes	Maximum memory available to the Java VM, in bytes.	OracleLogReader-Adapter
40006	EN	VM total memory allocated (B)	bytes	Total memory allocated to the Java VM at startup, in bytes.	OracleLogReader-Adapter
40007	EN	VM free memory (B)	bytes	Memory allocated but not used by the Java VM, in bytes.	OracleLogReader-Adapter
40008	EN	VM memory usage (B)	bytes	Memory allocated and in use by the Java VM, in bytes.	OracleLogReader-Adapter
40009	EN	VM % max memory used	percentage	Percentage of the maximum memory available to the Java VM, currently in use by the Java VM.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40010	EN	Current operation queue size	count	Current operation queue size.	OracleLogReader-Adapter
40011	EN	Log reposition point locator	n/a	Database transaction log locator the logreader uses to start to fetch records after resuming.	OracleLogReader-Adapter
40012	EN	Log reposition point LSN	n/a	Database transaction LSN (log file sequence number) the logreader uses to start to fetch records after resuming.	OracleLogReader-Adapter
40013	EN	Log reposition point SCN	n/a	Database transaction log SCN (system change number) the logreader uses to start to fetch records after resuming.	OracleLogReader-Adapter
40014	EN	Log reposition point timestamp	timestamp	Timestamp of the database transaction log record the logreader first fetched after resuming.	OracleLogReader-Adapter
40015	EN	Log reposition point timestamp (UTC)	timestamp	Timestamp of the database transaction log record the logreader first fetched after resuming (UTC).	OracleLogReader-Adapter
40016	EN	Last processed operation locator	n/a	Locator value of the most recently processed operation.	OracleLogReader-Adapter
40017	EN	Last processed operation LSN	n/a	LSN (Log sequence number) of the most recently processed operation.	OracleLogReader-Adapter
40018	EN	Last processed operation SCN	n/a	SCN (system change number) of the most recently processed operation.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40019	EN	Last processed operation timestamp	timestamp	Timestamp of the most recently processed operation.	OracleLogReaderAdapter
40020	EN	Last processed operation timestamp (UTC)	timestamp	Timestamp of the most recently processed operation (UTC).	OracleLogReaderAdapter
40021	EN	Average xlog operation wait time (ms)	milliseconds	Average wait time for getting a valid Log Reader processed operation since last reset, in milliseconds.	OracleLogReaderAdapter
40022	EN	Current scan buffer size	count	Current size of the Log Reader scan buffer.	OracleLogReaderAdapter
40023	EN	LogMiner Start Time (ms)	milliseconds	Total time spent starting LogMiner queries, in milliseconds.	OracleLogReaderAdapter
40024	EN	LogMiner Query Time (ms)	milliseconds	Total time spent executing LogMiner queries, in milliseconds.	OracleLogReaderAdapter
40025	EN	LogMiner Process Time (ms)	milliseconds	Total time spent reading the result set and moving data to the scan queue, in milliseconds.	OracleLogReaderAdapter
40026	EN	Average Records Per Scan	count	Average records returned per scan.	OracleLogReaderAdapter
40027	EN	Average Time Per Scan (ms)	milliseconds	Average time spent per scan, in milliseconds.	OracleLogReaderAdapter
40028	EN	Total LogMiner Scan Count	count	Total scans against LogMiner.	OracleLogReaderAdapter
40029	EN	Log records in each processed parallel scanner scan queue	count	Number of log records in each processed parallel scanner scan queue.	OracleLogReaderAdapter
40030	EN	LogMiner Scan Queue Wait Time (ms)	milliseconds	Total time waited to write the log records into parallel scan queue, in milliseconds.	OracleLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
40031	EN	LogMiner Scan Queue Poll Time (ms)	milliseconds	Total time scanners waited to poll the log records from parallel scan queue, in milliseconds.	OracleLogReaderAdapter
40032	EN	Total log records read	count	Number of log records read from LogMiner since the last start of the adapter instance.	OracleLogReaderAdapter
40033	EN	Total log records queued	count	Number of log records read from LogMiner and put into scan buffer for processing since the last start of the adapter instance.	OracleLogReaderAdapter
40034	EN	Total log records filtered	count	Number of log records read from LogMiner and not processed for any reason since the last start of the adapter instance.	OracleLogReaderAdapter
40035	EN	Last scanned record locator	n/a	Locator value of log record most recently retrieved from LogMiner.	OracleLogReaderAdapter
40036	EN	Last scanned record LSN	n/a	LSN (Log sequence number) log record most recently retrieved from LogMiner.	OracleLogReaderAdapter
40037	EN	Last scanned record SCN	n/a	SCN (system change number) log record most recently retrieved from LogMiner.	OracleLogReaderAdapter
40038	EN	Last scanned record timestamp	timestamp	Timestamp of log record most recently retrieved from LogMiner.	OracleLogReaderAdapter
40039	EN	Last scanned record timestamp (UTC)	timestamp	Timestamp of log record most recently retrieved from LogMiner (UTC).	OracleLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
40040	EN	Total operations scanned	count	Number of log records read from the scan queue and passed to the processor to generate operations since last reset.	OracleLogReaderAdapter
40041	EN	Total operations processed	count	Number of operations generated by the processor and passed to the sender to generate rowsets since last reset.	OracleLogReaderAdapter
40042	EN	Total operations skipped	count	Number of transaction log operations that weren't processed for any reason since last reset.	OracleLogReaderAdapter
40043	EN	Total maintenance user operations filtered	count	Number of maintenance user log records read from LogMiner and skipped since last reset.	OracleLogReaderAdapter
40044	EN	Average operation processing time (ms)	milliseconds	Average Log Reader operation processing time since last reset, in milliseconds.	OracleLogReaderAdapter
40045	EN	Total transactions processed	count	Number of transactions read from LogMiner and processed since last reset.	OracleLogReaderAdapter
40046	EN	Total transactions skipped	count	Number of transactions read from LogMiner and not processed for any reason since last reset.	OracleLogReaderAdapter
40047	EN	Total transactions opened	count	Number of begin transaction commands read from LogMiner since last reset.	OracleLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
40048	EN	Total transactions closed	count	Number of commit and rollback commands read from LogMiner since last reset.	OracleLogReader-Adapter
40049	EN	Total transactions committed	count	Number of commit commands read from LogMiner since last reset.	OracleLogReader-Adapter
40050	EN	Total transactions aborted (rolled back)	count	Number of rollback commands read from LogMiner since last reset.	OracleLogReader-Adapter
40051	EN	Total system transactions skipped	count	Number of system transactions read from LogMiner and skipped since last reset.	OracleLogReader-Adapter
40052	EN	Average operations per transaction	count	Average number of operations in each transaction read from LogMiner since last reset.	OracleLogReader-Adapter
40053	EN	Current session cache size	count	Current size of the session cache.	OracleLogReader-Adapter
40054	EN	Total LOB operations that processed by query data from PDB	count	Total LOB operations processed by querying data directly from the primary database.	OracleLogReader-Adapter
40055	EN	Average time that used to query PDB for LOB operation processing (ms)	milliseconds	Average time of LOB data query from primary database since last reset, in milliseconds.	OracleLogReader-Adapter
40056	EN	Current operation locator	n/a	Locator value of the log record most recently retrieved from the log scan queue.	OracleLogReader-Adapter
40057	EN	Current operation LSN	n/a	LSN (Log sequence number) value of the log record most recently retrieved from the log scan queue.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40058	EN	Current operation SCN	n/a	SCN (system change number) value of the log record most recently retrieved from the log scan queue.	OracleLogReader-Adapter
40059	EN	Current operation timestamp	timestamp	Timestamp of the log record most recently retrieved from the log scan queue.	OracleLogReader-Adapter
40060	EN	Current operation timestamp (UTC)	timestamp	Timestamp of the log record most recently retrieved from the log scan queue.	OracleLogReader-Adapter
40061	EN	Calls to standby scanner	count	Total times the standby scanner was invoked.	OracleLogReader-Adapter
40062	EN	Average calls to standby scanner per transaction	count	Average number of LogMiner scans issued by the standby scanner per transaction.	OracleLogReader-Adapter
40063	EN	Times to start LogMiner	count	Total number of LogMiner scans issued by the standby scanner.	OracleLogReader-Adapter
40064	EN	Total time spent by standby scanners starting LogMiner to load log records (ms)	milliseconds	Time spent starting LogMiner queries issued by the standby scanner, in milliseconds.	OracleLogReader-Adapter
40065	EN	Total time spent by standby scanners querying log records (ms)	milliseconds	Time spent executing LogMiner queries issued by the standby scanner, in milliseconds.	OracleLogReader-Adapter
40066	EN	Times to start LogMiner per transaction (ms)	milliseconds	Total number of LogMiner scans issued by the standby scanner for single transaction, in milliseconds.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40067	EN	Average time per standby scanner call (ms)	milliseconds	Average time spent per LogMiner scan issued by the standby scanner, in milliseconds.	OracleLogReaderAdapter
40068	EN	Rate of transactions that have unsupported operations	percentage	Total number of unsupported operations / total number of transactions that include unsupported operations.	OracleLogReaderAdapter
40069	EN	Log records cached by standby scanner	count	Number of log records cached by the standby scanner. In order to speed up processing of unsupported operations, the standby scanner tries to fetch more records than needed. If the next unsupported operation is located in this range, the result can be immediately returned. The range is equal to (expectedSCN, expectedSCN+5), and the cache will be cleaned up after commit.	OracleLogReaderAdapter
40070	EN	Total log records loaded by standby scanner	count	Total number of log records loaded by the standby scanner.	OracleLogReaderAdapter
40071	EN	Total opened database cursors	count	Total number of database cursors opened for LogMiner scan.	OracleLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
40072	EN	Calls to DDL standby scanner	count	Number of LogMiner scans issued by the DDL standby scanner. If this value keeps increasing, it can indicate that there are too many DDL change operations in the source. Because DDL standby scan is a time-consuming task, it can slow down replication.	OracleLogReaderAdapter
40073	EN	Total time spent by DDL standby scanners starting LogMiner to load log records (ms)	milliseconds	Time spent starting LogMiner queries issued by the DDL standby scanner, in milliseconds.	OracleLogReaderAdapter
40074	EN	Total time spent by DDL standby scanners querying log records (ms)	milliseconds	Time spent executing LogMiner queries issued by the DDL standby scanner, in milliseconds.	OracleLogReaderAdapter
40075	EN	Average time per DDL standby scanner call (ms)	milliseconds	Average time spent per LogMiner scan issued by the DDL standby scanner, in milliseconds.	OracleLogReaderAdapter
40076	EN	Total log records loaded by DDL standby scanner	count	Total number of log records loaded by the DDL standby scanner.	OracleLogReaderAdapter
40077	EN	Total opened database cursors for DDL standby scanner	count	Total number of database cursors opened for DDL standby scanner.	OracleLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
40078	EN	Total persisted open transactions	count	Total number of persisted open transactions. In order to more timely move the truncation point, the transaction context needs to be persisted into the repository each hour or every 100000 transactions.	OracleLogReader-Adapter
40079	EN	Total open transactions persistence time (ms)	milliseconds	Time spent persisting transaction context into repository, in milliseconds.	OracleLogReader-Adapter
40080	EN	Last time to persist open transactions (ms)	milliseconds	Most recent amount of time spent persisting transaction context into the repository, in milliseconds.	OracleLogReader-Adapter
40081	EN	Time open transaction persistence last started	count	Most recent timestamp that transaction context was persisted into the repository.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40082	EN	Times to start scanner with COMMITTED_DATA_ONLY mode to handle transaction with rollback to save point	count	Number of times to start scanner with COMMITTED_DATA_ONLY mode to handle transaction with rollback to save point. In order to speed up rollback to save point handling, additional scanners start to fetch transactions including the rollback to save point records in COMMITTED_DATA_ONLY mode. This way, only committed records are returned, and the rollback to save point records is safely skipped.	OracleLogReaderAdapter
40083	EN	Total time rollback rescan scanner to start LogMiner and load log records (ms)	milliseconds	Time spent starting LogMiner queries issued by rollback rescan scanner, in milliseconds.	OracleLogReaderAdapter
40084	EN	Total time rollback rescan scanner to process the log records load from LogMiner in COMMITTED_DATA_ONLY mode (ms)	milliseconds	Time spent executing LogMiner queries and handling the schema changes during rollback rescan scan, in milliseconds.	OracleLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
40085	EN	Current Op Proc RASD marked object cache size	count	Current marked object cached in the marked object ID cache. This cache is used to speed up fetching metadata by object ID for tables marked for replication. By default, the max cache size is 1000, so in most cases the marked object can be found in the cache.	OracleLogReader-Adapter
40086	EN	Total Op Proc RASD marked object cache hits	count	Total number of operation processor marked object repository ID cache hits.	OracleLogReader-Adapter
40087	EN	Total Op Proc RASD marked object cache misses	count	Total number of operation processor marked object repository ID cache misses.	OracleLogReader-Adapter
40088	EN	Current Op Proc_Name RASD marked object cache size	count	Current marked object cached in the marked object ID cache. This cache is used to speed up fetching metadata by object ID for tables marked for replication. By default, the max cache size is 1000, so in most cases the marked object can be found in the cache.	OracleLogReader-Adapter
40089	EN	Total Op Proc_Name RASD marked object cache hits	count	Total number of operation processor marked object repository name cache hits.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40090	EN	Total Op Proc_Name RASD marked object cache misses	count	Total number of operation processor marked object repository name cache misses.	OracleLogReader-Adapter
40091	EN	Average sender operation processing time (ms)	milliseconds	Average sender operation processing time, in milliseconds.	OracleLogReader-Adapter
40092	EN	Average sender operation wait time (ms)	milliseconds	Average sender operation wait time, in milliseconds.	OracleLogReader-Adapter
40093	EN	Average change set send time (ms)	milliseconds	Average change set send time, in milliseconds.	OracleLogReader-Adapter
40094	EN	Sender operations processed	count	Number of sender operations processed.	OracleLogReader-Adapter
40095	EN	Current marked objects cache size	count	Current marked objects cache size. The default value is 1000.	OracleLogReader-Adapter
40096	EN	Remote_Source_Processed_Row_LSN	n/a	LSN (Log sequence number) value of most recent operation processed by the adapter. NOTE: Same as Last processed operation LSN.	OracleLogReader-Adapter
40097	EN	Remote_Source_Processed_Row_Timestamp	timestamp	Timestamp of the most recently processed operation.	OracleLogReader-Adapter
40098	EN	Remote_Source_Current_Row_LSN	n/a	Current timestamp of Oracle database (UTC).	OracleLogReader-Adapter
40099	EN	Remote_Source_Current_Row_Timestamp	timestamp	Log sequence number of CURRENT archive log file information.	OracleLogReader-Adapter
40100	EN	Average row set send time (ms)	milliseconds	Average row set send time, in milliseconds.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40101	EN	Sender operations processed	count	Number of sender operations processed.	OracleLogReader-Adapter
40102	EN	Row sets sent	count	Number of row sets sent by agent to server.	OracleLogReader-Adapter
40103	EN	Average row sets per second	count	Average number of row sets sent per second to DPServer since last reset.	OracleLogReader-Adapter
40104	EN	Last sequence ID sent	n/a	Last sequence ID of row set sent to DPServer.	OracleLogReader-Adapter
40105	EN	Last transaction ID sent	n/a	Last transaction ID of row set sent to DPServer.	OracleLogReader-Adapter
40106	EN	Begin marker received	timestamp	Timestamp when the adapter received the begin marker request from DPServer.	OracleLogReader-Adapter
40107	EN	Begin marker applied	timestamp	Timestamp when the begin marker was applied to the source.	OracleLogReader-Adapter
40108	EN	Begin marker sent	timestamp	Timestamp when the begin marker was read from the source and sent to DPServer.	OracleLogReader-Adapter
40109	EN	End marker received	timestamp	Timestamp when the end marker was received by the adapter from DPServer.	OracleLogReader-Adapter
40110	EN	End marker applied	timestamp	Timestamp when the end marker was applied to the source.	OracleLogReader-Adapter
40111	EN	End marker sent	timestamp	Timestamp when the end marker was read from the source and sent to DPServer.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40112	EN	Invalid scans	count	Cumulative number of invalid scans, starting from the most recent start or resume of the adapter instance. If this number is increasing, there are no changes to any of the source tables of a remote source.	OracleLogReader-Adapter
40113	EN	Average time to scan trigger queue (ms)	milliseconds	Average time taken to scan the trigger queue, in milliseconds, calculated from the most recent start or resume of the adapter instance. This value is calculated for the active processing time, and doesn't include time when the adapter instance was idle. (Total cumulative scan time/Total scans)	OracleLogReader-Adapter
40114	EN	Total trigger queue records scanned	count	Total number of records scanned by the OracleLogReaderAdapter for a remote source.	OracleLogReader-Adapter
40115	EN	Total scan batches	count	In each scan, trigger queue records are split into batches based on table name and max batch size.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40116	EN	Average records per scan	count	If the number of unscanned trigger queue records is large and the average records per scan is much smaller than the maximum scan size configured in the remote source, there are too many small transactions per scan. In this case, increase the maximum transaction count per scan in the remote source configuration.	OracleLogReader-Adapter
40117	EN	Average records per batch	count	Average number of records per batch: (Total number of trigger queue records scanned/Total number of scan batches)	OracleLogReader-Adapter
40118	EN	Average time to scan trigger queue, per record (ms)	milliseconds	Average time taken per record to retrieve the trigger queue and split into different batches, in milliseconds.	OracleLogReader-Adapter
40119	EN	Total transactions	count	Total number of transactions processed.	OracleLogReader-Adapter
40120	EN	Unscanned trigger queue records	count	Current number of unscanned trigger queue records. If this value is decreasing, replication is catching up with the source changes. Otherwise, replication can lag.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40121	EN	Latest scanned trans_time in trigger queue	timestamp	Latest scanned trans_time in the trigger queue. If this value is closer to "Latest trans_time in trigger queue", then there's no lag in scanning.	OracleLogReader-Adapter
40122	EN	Latest trans_time in trigger queue	timestamp	Latest trans_time (that is, max) in the trigger queue. If this value changes continuously, then there are changes to source tables in this remote source.	OracleLogReader-Adapter
40123	EN	Max scan number in trigger queue	number	Max scan number in trigger queue.	OracleLogReader-Adapter
40124	EN	Times shadow table results retrieved	count	Number of times the shadow table results have been retrieved.	OracleLogReader-Adapter
40125	EN	Average time to retrieve shadow table results (ms)	milliseconds	If this value is increasing over time, it indicates that the source database may be slow to respond.	OracleLogReader-Adapter
40126	EN	Total rowsets sent out	count	A row set can have multiple rows up to max batch size.	OracleLogReader-Adapter
40127	EN	Total rows sent out	count	Total number of rows sent out.	OracleLogReader-Adapter
40128	EN	Average rows per rowset	count	Average number of rows per rowset: (Total count of rows sent out/ Total count of rowsets sent out). When there are many small transactions, this value is closer to 1.0.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40129	EN	Average time to send rowsets (ms)	milliseconds	Average time to send rowsets DPServer, in milliseconds.	OracleLogReader-Adapter
40130	EN	Latest timestamp applier sent rows	timestamp	Most recent time the adapter's applier sent rows to DPServer.	OracleLogReader-Adapter
40131	EN	Latest timestamp applier sent rows (UTC)	n/a	Most recent time the adapter's applier sent rows to DPServer (UTC).	OracleLogReader-Adapter
40132	EN	Latest applier sent sequence ID	timestamp	Sequence ID of the last row the adapter's applier has sent to DPServer.	OracleLogReader-Adapter
40133	EN	Latest received truncated sequence ID	timestamp	Sequence ID of the last row received by the DPServer.	OracleLogReader-Adapter
40134	EN	Log reposition point thread	number	Oracle redo log thread from which the LogReader starts to fetch records after remote source is resumed.	OracleLogReader-Adapter
40135	EN	Last processed operation thread	number	Oracle redo log thread of the last processed operation.	OracleLogReader-Adapter
40136	EN	Last scanned record thread	number	Oracle redo log thread of the last record retrieved from LogMiner.	OracleLogReader-Adapter
40137	EN	Current operation thread	number	Oracle redo log thread of the most recent log retrieved from the LogReader scan queue.	OracleLogReader-Adapter
40138	EN	Remote_Source_Processed_Row_Thread	number	Oracle redo log thread of last processed record of the remote source.	OracleLogReader-Adapter
40139	EN	Remote_Source_Current_Row_Thread	number	Oracle redo log thread of CURRENT redo log file.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40140	EN	Total redo log files read	number	Total number of logs read from LogMiner since remote source was resumed.	OracleLogReader-Adapter
40141	EN	DDL transaction with longest processing time		DDL transaction taking the longest time to be processed in SDI. This statistic is valuable in both scenarios when "Enable rescan for DDL operations" is either true or false.	OracleLogReader-Adapter
40142	EN	DDL transactions with long processing times		List of DDL transactions taking a long time to be processed in SDI and beyond the threshold such as 1 hour. When "Enable rescan for DDL operations" is false, DDL transactions with long processing times don't block the consequent DML transactions. When "Enable rescan for DDL operations" is true, DDL transactions with long processing times block consequent DML transactions and throw an exception when the timeout threshold is reached.	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40143	EN	Total scans	count	Cumulative number of total scans, starting from the most recent start or resume of the adapter instance. This number should be increasing if any of the remote source tables have continuous changes.	OracleLogReader-Adapter
40144	EN	Max scan time per record over the latest statistics update interval (ms)	milliseconds	If this value is larger than 0 seconds, the source database may be busy.	OracleLogReader-Adapter
40145	EN	Trigger queue records in the slowest scan	count	Number of trigger queue records in the slowest scan over the latest statistics update interval (MAX_SCAN_TIME_PER_RECORD equals the value above).	OracleLogReader-Adapter
40146	EN	Min scan time per record over latest statistics update interval (ms)	milliseconds	If this value is larger than 0 seconds, the source database may be busy.	OracleLogReader-Adapter
40147	EN	Trigger queue records in the fastest scan	count	Number of trigger queue records in the fastest scan over the latest statistics update interval (MIN_SCAN_TIME_PER_RECORD equals the value above).	OracleLogReader-Adapter

ID	Language	Statistic Name	Unit	Long Description	Component
40148	EN	Records in shadow table	count	Value string: <number of records in shadow table>;<remotetableID>;"<shadow table schema>". "<shadow table name>"	OracleLogReaderAdapter
40149	EN	Average transaction size	count	Overall average transaction size since the remote source started.	OracleLogReaderAdapter
40150	EN	Dispersion of transaction size	count	Standard variance of transaction size since the remote source started.	OracleLogReaderAdapter
50001	EN	Time statistics obtained	timestamp	Time when statistics were obtained from the LogReader.	MssqlLogReaderAdapter
50002	EN	Time replication last started	timestamp	Most recent time LogReader started to replicate data.	MssqlLogReaderAdapter
50003	EN	Time statistics last reset	timestamp	Most recent time statistics counters were reset in the LogReader.	MssqlLogReaderAdapter
50004	EN	Items in Global LRUCache	count	Number of object references in the internal Least Recently Used cache.	MssqlLogReaderAdapter
50005	EN	VM maximum memory (B)	bytes	Maximum available memory of Java VM, in bytes.	MssqlLogReaderAdapter
50006	EN	VM total memory allocated (B)	bytes	Current memory allocated for Java VM, in bytes.	MssqlLogReaderAdapter
50007	EN	VM free memory (B)	bytes	Free memory of Java VM, in bytes.	MssqlLogReaderAdapter
50008	EN	VM memory usage (B)	bytes	Current used memory of Java VM, in bytes.	MssqlLogReaderAdapter
50009	EN	VM % max memory used	percentage	Used memory percentage of Java VM.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50010	EN	Current operation queue size	count	Current number of operations in the operation queue. A low value can suggest a performance bottleneck in the operation processor component. A high value can suggest a performance bottleneck in downstream components like sender.	MssqlLogReaderAdapter
50011	EN	Log reposition point locator	n/a	Start locator of log scanning. If the LogReader is suspended or halted, it starts scanning data from the transaction log by the LSN extracted from this locator next time LogReader is started.	MssqlLogReaderAdapter
50012	EN	Log reposition point LSN	n/a	Start LSN of log scanning. If the LogReader is suspended or halted, it will start scan data from MSSQL transaction log by this LSN next time LogReader is started.	MssqlLogReaderAdapter
50013	EN	Last processed operation locator	n/a	Locator of the log record last processed by the scanner. It indicates the position in the MSSQL transaction log where LogReader is processing.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50014	EN	Last processed operation LSN	n/a	LSN of the log record last processed by the scanner. It indicates the position in the MSSQL transaction log where LogReader is processing.	MssqlLogReaderAdapter
50015	EN	Last processed transaction begin time	timestamp	Transaction begin time of the transaction last processed by the scanne.r	MssqlLogReaderAdapter
50016	EN	Last processed transaction begin time (UTC)	timestamp	Transaction begin time of the transaction last processed by the scanner (UTC).	MssqlLogReaderAdapter
50017	EN	Remote_Source_Processed_Row_LSN	n/a	LSN of the log record last processed by the scanner. It indicates the position in the MSSQL transaction log where LogReader is processing.	MssqlLogReaderAdapter
50018	EN	Remote_Source_Processed_Row_Timestamp	timestamp	Transaction begin time of the transaction last processed by the scanner (UTC).	MssqlLogReaderAdapter
50019	EN	Remote_Source_Current_Row_LSN	n/a	LSN of the log record currently being processed by the scanner.	MssqlLogReaderAdapter
50020	EN	Remote_Source_Current_Row_Timestamp	timestamp	Transaction begin time of the transaction currently being processed by the scanner (UTC).	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50021	EN	Average xlog operation wait time (ms)	milliseconds	Average wait time when requesting a log record from scanner queue, in milliseconds. A high value can suggest a performance problem in the scanner.	MssqlLogReaderAdapter
50022	EN	Total operations scanned	count	Number of operations scanned from MSSQL transaction log.	MssqlLogReaderAdapter
50023	EN	Total operations processed	count	Number of operations outputted by Operation Processor.	MssqlLogReaderAdapter
50024	EN	Total operations skipped	count	Number of operations skipped because of out of scan range of the nested operation processor.	MssqlLogReaderAdapter
50025	EN	Total maintenance user operations filtered	count	Number of filtered operations performed by configured maintenance user.	MssqlLogReaderAdapter
50026	EN	Average operation processing time (ms)	milliseconds	Average time to process each operation by the operation processor, in milliseconds. A high value can suggest a performance problem in the operation processor.	MssqlLogReaderAdapter
50027	EN	Total transactions processed	count	Number of BEGIN_TRANSACTION operations processed by the operation processor.	MssqlLogReaderAdapter
50028	EN	Total transactions skipped	count	Number of transactions rolled back or skipped by the operation processor.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50029	EN	Total transactions opened	count	Same as Total transactions processed.	MssqlLogReaderAdapter
50030	EN	Total transactions closed	count	Number of COMMIT and ROLLBACK operations processed by the operation processor.	MssqlLogReaderAdapter
50031	EN	Total transactions committed	count	Number of COMMIT operations processed by the operation processor.	MssqlLogReaderAdapter
50032	EN	Total transactions aborted (rolled back)	count	Number of ROLLBACK operations processed by the operation processor.	MssqlLogReaderAdapter
50033	EN	Average ops per transaction	count	Average transaction size processed by the operation processor.	MssqlLogReaderAdapter
50034	EN	Current scan buffer size	count	Current number of log records in the scanner queue. A low value can suggest a performance bottleneck in the scanner component. A high value can suggest a performance bottleneck in downstream components like operation processor and sender.	MssqlLogReaderAdapter
50035	EN	Current LCT cache size	count	Current cache size of Last Committed Transaction. LCT cache is used to construct log truncation command parameters.	MssqlLogReaderAdapter
50036	EN	Total bytes read (B)	bytes	Bytes read from MSSQL transaction log by the scanner.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50037	EN	Total log read time (ms)	milliseconds	Time spent reading MSSQL log file by the scanner, in milliseconds.	MssqlLogReaderAdapter
50038	EN	Average time (ms) per log device read	milliseconds	Average time spent each time the scanner attempts to read the log file, in milliseconds. A high value can suggest a performance problem during transaction log file reading itself, such as an I/O throughput problem or network problem when reading a remote MSSQL log file by file sharing.	MssqlLogReaderAdapter
50039	EN	Average bytes read per second (B)	bytes	Scanner log reading speed, in bytes per second. A low value can suggest a performance problem during transaction log file reading itself, such as an I/O throughput problem or network problem when reading a remote MSSQL log file by file sharing.	MssqlLogReaderAdapter
50040	EN	Current blocks per read	count	Current block count of each bulk reading by scanner.	MssqlLogReaderAdapter
50041	EN	Estimated time to the end of the log (s)	seconds	Estimated time to the end of the log, in seconds. Disabled by default for performance considerations. It invokes select max([Current LSN]) from fn_dblog(NULL,NULL).	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50042	EN	Approximate log operations to scan	count	Approximate number of log operations to scan. Disabled by default for performance considerations. It invokes select max([Current LSN]) from fn_dblog(NULL,NULL).	MssqlLogReaderAdapter
50043	EN	Average sender operation processing time (ms)	milliseconds	Average time to process each change data operation between when it's taken from LogReader output and put into sender queue after being processed by the sender to construct adapter CDC rows, in milliseconds. A high value can suggest a performance problem in the sender.	MssqlLogReaderAdapter
50044	EN	Average sender operation wait time (ms)	milliseconds	Average time waited when offering a change data operation taken from LogReader to the sender processing queue, in milliseconds. A high value can suggest a performance problem in sender.	MssqlLogReaderAdapter
50045	EN	Average change set send time (ms)	milliseconds	Average change set send time, in milliseconds.	MssqlLogReaderAdapter
50046	EN	Sender operations processed	count	Number of change data operations from LogReader processed by sender.	MssqlLogReaderAdapter
50047	EN	Current marked objects cache size	count	Current marked objects cache size.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50048	EN	Log truncation occurrences	count	Number of log truncation occurrences.	MssqlLogReaderAdapter
50049	EN	Total log truncation requests	count	Total number of log truncation requests.	MssqlLogReaderAdapter
50050	EN	Total log truncation requests ignored because the truncation point didn't change	count	Total number of log truncation requests ignored because the truncation point didn't change.	MssqlLogReaderAdapter
50051	EN	Total log truncation requests ignored because the specified LCT LSN wasn't found	count	Total number of log truncation requests ignored because the specified LCT LSN wasn't found.	MssqlLogReaderAdapter
50052	EN	Total truncation time (ms)	milliseconds	Total time spent in log truncation, in milliseconds.	MssqlLogReaderAdapter
50053	EN	Total execution time of sp_repldone (ms)	milliseconds	Total execution time of sp_repldone, which is the MSSQL built-in function used for log truncation, in milliseconds.	MssqlLogReaderAdapter
50054	EN	Total truncation LCT lookup time (ms)	milliseconds	Total time spent in LCT (Last Committed Transaction) lookup, which is used to construct parameters passed to log truncation command sp_repldone, in milliseconds.	MssqlLogReaderAdapter
50055	EN	Total primary database log truncated (B)	bytes	Total bytes in MSSQL transaction log truncated.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50056	EN	Current LCT LSN	n/a	Current LSN of the Last Committed Transaction, which is used to construct parameters passed to log truncation command sp_repldone.	MssqlLogReaderAdapter
50057	EN	Last MS log truncation - Command	n/a	Generated MSSQL command for log truncation last time.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50058	EN	Last MS log truncation - Status	n/a	Status of the last log truncation. • *SUCCESS*: Log truncation command has been successfully executed by MSSQL. • *RUNNING*: Log truncation is activated and is running. • *LCT_LOOKUP_MISS*: LCT (Last Committed Transaction) lookup failed. Log truncation is skipped because the LCT required to fill the parameters of log truncation command sp_repldone isn't available in the agent cache. Typically, this is a transit situation in a short time period after DPAgent is restarted. It will disappear after a short time when the LCT cache is established. • *SQL_EXECUTION_FAIL*: Log truncation fails because sp_repldone fails to exe-	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
				cute on the MSSQL side. Check the adapter logs for details. • *IGNORE_OLDER_OAT*: Log truncation is skipped because the truncation point is older than the truncation point used in last successful log truncation. This means the MSSQL transaction log content before the current truncation point has already been truncated in previous log truncations. • *FAIL*: Log truncation fails for other reasons. Check adapter logs for details.	
50059	EN	Last MS log truncation - Timestamp	timestamp	Timestamp when the last MSSQL log truncation happened.	MssqlLogReaderAdapter
50060	EN	Last MS log truncation - Saved truncation point	n/a	Log truncation point used in previous log truncation.	MssqlLogReaderAdapter
50061	EN	Last MS log truncation - Current truncation point	n/a	Log truncation point picked for current log truncation.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50062	EN	Last truncation point update - Timestamp	timestamp	Timestamp when last time log truncation point is updated to LogReader.	MssqlLogReaderAdapter
50063	EN	Last truncation point update - Locator	n/a	Last updated log truncation point.	MssqlLogReaderAdapter
50064	EN	Average row set send time (ms)	milliseconds	Average time to add a row set from sender to agent response queue, in milliseconds. A high value can suggest a performance problem in agent serialization, network, or DPServer side.	MssqlLogReaderAdapter
50065	EN	Row sets sent	count	Number of row sets sent by sender. Processed change data operations are grouped and sent as row sets.	MssqlLogReaderAdapter
50066	EN	Average row sets per second	count	Average number of row sets processed by sender per second. A low value can suggest a performance problem in sender or downstream components.	MssqlLogReaderAdapter
50067	EN	Last sequence ID sent	n/a	Sequence ID of the last sent CDC row.	MssqlLogReaderAdapter
50068	EN	Last transaction ID sent	n/a	Last sent transaction ID.	MssqlLogReaderAdapter
50069	EN	Subscribed tables	count	Number of subscribed tables.	MssqlLogReaderAdapter
50070	EN	Last connection time	timestamp	Time last connected to the source database.	MssqlLogReaderAdapter
50071	EN	Occupied log size as a percent of the total log size	percentage	Percentage of the source database log space used.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50072	EN	Uncommitted transactions of the current used database	count	Open transaction count in the source database.	MssqlLogReaderAdapter
50073	EN	Server instance hosting the current primary replica	n/a	Name of the server instance that is hosting the current primary replica in MSSQL AlwaysOn environment.	MssqlLogReaderAdapter
50074	EN	Expiration date for the current logged in user	timestamp	Expiration date for the currently logged in user.	MssqlLogReaderAdapter
50079	EN	Begin marker received	timestamp	Timestamp when the adapter received the begin marker request from DPServer.	MssqlLogReaderAdapter
50080	EN	Begin marker applied	timestamp	Timestamp when the begin marker was applied to the source.	MssqlLogReaderAdapter
50081	EN	Begin marker sent	timestamp	Timestamp when the begin marker was read from the source and sent to DPServer.	MssqlLogReaderAdapter
50082	EN	End marker received	timestamp	Timestamp when the end marker was received by the adapter from DPServer.	MssqlLogReaderAdapter
50083	EN	End marker applied	timestamp	Timestamp when the end marker was applied to the source.	MssqlLogReaderAdapter
50084	EN	End marker sent	timestamp	Timestamp when the end marker was read from the source and sent to DPServer.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50085	EN	Invalid scans	count	Cumulative number of invalid scans, starting from the most recent start or resume of the adapter instance. If this number is increasing, there are no changes to any of the source tables of a remote source.	MssqlLogReaderAdapter
50086	EN	Average time to scan the trigger queue (ms)	milliseconds	Average time taken to scan the trigger queue, in milliseconds, calculated from the most recent start or resume of the adapter instance. This value is calculated for the active processing time, and doesn't include time when the adapter instance was idle. (Total cumulative scan time/Total scans)	MssqlLogReaderAdapter
50087	EN	Total trigger queue records scanned	count	Total number of records scanned by MssqlLogReaderAdapter for a remote source.	MssqlLogReaderAdapter
50088	EN	Total scan batches	count	In each scan, trigger queue records are split into batches based on table name and max batch size.	MssqlLogReaderAdapter
50089	EN	Average trigger queue records per scan	count	Average number of trigger queue records per scan.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50090	EN	Average records per batch	count	Average number of records per batch: (Total number of trigger queue records scanned/Total number of scan batches)	MssqlLogReaderAdapter
50091	EN	Average time to scan trigger queue, per record (ms)	milliseconds	Average time taken per record to retrieve the trigger queue and split into different batches, in milliseconds.	MssqlLogReaderAdapter
50092	EN	Total transactions	count	Total number of transactions processed.	MssqlLogReaderAdapter
50093	EN	Unscanned trigger queue records	count	Current number of unscanned trigger queue records. If this value is decreasing, replication is catching up with the source changes. Otherwise, replication can lag.	MssqlLogReaderAdapter
50094	EN	Latest scanned trans_time in trigger queue	timestamp	Latest scanned trans_time in the trigger queue. If this value is closer to “Latest trans_time in trigger queue”, then there's no lag in scanning.	MssqlLogReaderAdapter
50095	EN	Latest trans_time in trigger queue	timestamp	Latest trans_time (that is, max) in the trigger queue. If this value changes continuously, then there are changes to source tables in this remote source.	MssqlLogReaderAdapter
50096	EN	Max scan number in trigger queue	number	Max scan number in trigger queue.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50097	EN	Times shadow table results retrieved	count	Number of times the shadow table results have been retrieved.	MssqlLogReaderAdapter
50098	EN	Average time to retrieve shadow table results (ms)	milliseconds	If this value is increasing over time, it indicates that the source database may be slow to respond.	MssqlLogReaderAdapter
50099	EN	Total rowsets sent out	count	A row set can have multiple rows up to max batch size.	MssqlLogReaderAdapter
50100	EN	Total rows sent out	count	Total number of rows sent out.	MssqlLogReaderAdapter
50101	EN	Average rows per rowset	count	Average number of rows per rowset: (Total count of rows sent out/ Total count of rowsets sent out). When there are many small transactions, this value is closer to 1.0.	MssqlLogReaderAdapter
50102	EN	Average time to send rowsets (ms)	milliseconds	Average time to send rowsets to DPServer, in milliseconds.	MssqlLogReaderAdapter
50103	EN	Latest timestamp applier sent rows	timestamp	Most recent time the adapter's applier sent rows to DPServer.	MssqlLogReaderAdapter
50104	EN	Latest timestamp applier sent rows (UTC)	timestamp	Most recent time the adapter's applier sent rows to DPServer (UTC).	MssqlLogReaderAdapter
50105	EN	Latest applier sent sequence ID	timestamp	Sequence ID of the last row the adapter's applier has sent to DPServer.	MssqlLogReaderAdapter
50106	EN	Latest received truncated sequence ID	timestamp	Sequence ID of the last row received by the DPServer.	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50107	EN	Total scans	count	Cumulative number of total scans, starting from the most recent start or resume of the adapter instance. This number should be increasing if any of the remote source tables have continuous changes.	MssqlLogReaderAdapter
50108	EN	Max scan time per record over the latest statistics update interval (ms)	milliseconds	If this value is larger than 0 seconds, the source database may be busy.	MssqlLogReaderAdapter
50109	EN	Trigger queue records in the slowest scan	count	Number of trigger queue records in the slowest scan over the latest statistics update interval (MAX_SCAN_TIME_PER_RECORD equals the value above).	MssqlLogReaderAdapter
50110	EN	Min scan time per record over latest statistics update interval (ms)	milliseconds	If this value is larger than 0 seconds, the source database may be busy.	MssqlLogReaderAdapter
50111	EN	Trigger queue records in the fastest scan	count	Number of trigger queue records in the fastest scan over the latest statistics update interval (MIN_SCAN_TIME_PER_RECORD equals the value above).	MssqlLogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
50112	EN	Records in shadow table	count	Value string: <number of records in shadow table>;<remotetableID>;"<shadow table schema>". "<shadow table name>"	MssqlLogReaderAdapter
50113	EN	Average transaction size	count	Overall average transaction size since the remote source started.	MssqlLogReaderAdapter
50114	EN	Dispersion of transaction size	count	Standard variance of transaction size since the remote source started.	MssqlLogReaderAdapter
60001	EN	Time statistics obtained	timestamp	Day, date, and time when statistics were obtained.	DB2LogReaderAdapter
60002	EN	Time replication last started	timestamp	Day, date, and time that Replicating state started.	DB2LogReaderAdapter
60003	EN	Time statistics last reset	timestamp	Day, date, and time that statistics counters were reset.	DB2LogReaderAdapter
60004	EN	Items held in Global LRUCache	count	Number of object references in the internal Least Recently Used cache.	DB2LogReaderAdapter
60005	EN	VM maximum memory (B)	bytes	Maximum memory available to the Java VM, in bytes.	DB2LogReaderAdapter
60006	EN	VM total memory allocated (B)	bytes	Total memory allocated to the Java VM at startup, in bytes.	DB2LogReaderAdapter
60007	EN	VM free memory (B)	bytes	Memory allocated but not used by the Java VM, in bytes.	DB2LogReaderAdapter
60008	EN	VM memory usage (B)	bytes	Memory allocated and in use by the Java VM, in bytes.	DB2LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
60009	EN	VM % max memory used	percentage	Percentage of the maximum memory available to the Java VM, currently in use by the Java VM.	DB2LogReaderAdapter
60010	EN	XLog scans	count	Number of XLog scans.	DB2LogReaderAdapter
60011	EN	Average unprocessed operations per XLog scan	count	Average number of operations returned per scan.	DB2LogReaderAdapter
60012	EN	Average XLog scan time (s)	seconds	Average time to scan transaction log, in seconds.	DB2LogReaderAdapter
60013	EN	Operations replicated	count	Number of operations replicated, excluding BEGIN/COMMIT/ROLLBACK operations.	DB2LogReaderAdapter
60014	EN	Transactions replicated	count	Number of transactions replicated.	DB2LogReaderAdapter
60015	EN	XLog operations skipped (maint_user, unmarked tables)	count	Number of Maint User transactions skipped (valid only if send_maint_user_trxs_to_replicate = true), and the number of transaction log rows referencing unmarked tables skipped.	DB2LogReaderAdapter
60016	EN	Average wait time on empty Xlog (ms)	milliseconds	Average time waiting for transaction log data, in milliseconds.	DB2LogReaderAdapter
60017	EN	Average PDB Service Time / Operation	n/a	Average transaction log operation arrival rate: (average XLog access times) / number of operations processed.	DB2LogReaderAdapter
60018	EN	Operation Queue Size	count	Size of the operation queue.	DB2LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
60019	EN	Operation Data Hash Size	count	Size of the operation data hash.	DB2LogReaderAdapter
60020	EN	Transactions truncated	count	Number of transactions truncated.	DB2LogReaderAdapter
60021	EN	Current operation queue size	count	Current number of operations in the operation queue. A low value may suggest a performance bottleneck in the operation processor component. A high value may suggest a performance bottleneck in downstream components like sender.	DB2LogReaderAdapter
60022	EN	Log reposition point locator	n/a	Position in the log that the scan will start from.	DB2LogReaderAdapter
60023	EN	Log reposition point LSN	n/a	LSN in the log that the scan will start from.	DB2LogReaderAdapter
60024	EN	Last processed operation LSN	n/a	LSN of the last processed operation.	DB2LogReaderAdapter
60025	EN	Total operation process time (s)	seconds	Time used by the UDB Operation Processor, in seconds.	DB2LogReaderAdapter
60026	EN	Average sender operation processing time (ms)	milliseconds	Average time to send an operation, in milliseconds.	DB2LogReaderAdapter
60027	EN	Average sender operation wait time (ms)	milliseconds	Average time between sending operations, in milliseconds.	DB2LogReaderAdapter
60028	EN	Average row set send time (ms)	milliseconds	Average time to send a row set, in milliseconds.	DB2LogReaderAdapter
60029	EN	Sender operations processed	count	Number of operations sent.	DB2LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
60030	EN	Row sets sent	count	Number of row sets sent by sender. Processed change data operations are grouped and sent as row sets.	DB2LogReaderAdapter
60031	EN	Average row sets per second	count	Average number of row sets processed by sender per second. A low value may suggest a performance problem in sender or downstream components.	DB2LogReaderAdapter
60032	EN	Last sequence ID sent	n/a	Sequence ID of the last CDC row sent.	DB2LogReaderAdapter
60033	EN	Last transaction ID sent	n/a	Transaction ID of the last row set sent to data provisioning server.	DB2LogReaderAdapter
60034	EN	Begin marker received	timestamp	Timestamp when the adapter received the begin marker request from DPServer.	DB2LogReaderAdapter
60035	EN	Begin marker applied	timestamp	Timestamp when the begin marker was applied to the source.	DB2LogReaderAdapter
60036	EN	Begin marker sent	timestamp	Timestamp when the begin marker was read from the source and sent to DPServer.	DB2LogReaderAdapter
60037	EN	End marker received	timestamp	Timestamp when the end marker was received by the adapter from DPServer.	DB2LogReaderAdapter
60038	EN	End marker applied	timestamp	Timestamp when the end marker was applied to the source.	DB2LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
60039	EN	End marker sent	timestamp	Timestamp when the end marker was read from the source and sent to DPServer.	DB2LogReaderAdapter
60040	EN	Current scan queue size	count	Current scan queue size.	DB2LogReaderAdapter
60041	EN	Invalid scans	count	Cumulative number of invalid scans, starting from the most recent start or resume of the adapter instance. If this number is increasing, there are no changes to any of the source tables of a remote source.	DB2LogReaderAdapter
60042	EN	Average time to scan the trigger queue (ms)	milliseconds	Average time taken to scan the trigger queue, in milliseconds, calculated from the most recent start or resume of the adapter instance. This value is calculated for the active processing time, and does not include time when the adapter instance was idle. (Total cumulative scan time/Total scans)	DB2LogReaderAdapter
60043	EN	Total trigger queue records scanned	count	Total number of records scanned by DB2LogReaderAdapter for a remote source.	DB2LogReaderAdapter
60044	EN	Total scan batches	count	In each scan, trigger queue records are split into batches based on table name and max batch size.	DB2LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
60045	EN	Average trigger queue records per scan	count	Average number of trigger queue records per scan.	DB2LogReaderAdapter
60046	EN	Average records per batch	count	Average number of records per batch: (Total number of trigger queue records scanned/Total number of scan batches)	DB2LogReaderAdapter
60047	EN	Average time to scan trigger queue, per record (ms)	milliseconds	Average time taken per record to retrieve the trigger queue and split into different batches, in milliseconds.	DB2LogReaderAdapter
60048	EN	Total transactions	count	Total number of transactions processed.	DB2LogReaderAdapter
60049	EN	Unscanned trigger queue records	count	Current number of unscanned trigger queue records. If this value is decreasing, replication is catching up with the source changes. Otherwise replication may lag	DB2LogReaderAdapter
60050	EN	Latest scanned trans_time in trigger queue	timestamp	Latest scanned trans_time in the trigger queue. If this value is closer to 'Latest trans_time in trigger queue' then there is no lag in scanning.	DB2LogReaderAdapter
60051	EN	Latest trans_time in trigger queue	timestamp	Latest trans_time (i.e. max) in the trigger queue. If this value changes continuously then there are changes to source tables in this remote source.	DB2LogReaderAdapter
60052	EN	Max scan number in trigger queue	number	Max scan number in trigger queue.	DB2LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
60053	EN	Times shadow table results retrieved	count	Number of times the shadow table results have been retrieved.	DB2LogReaderAdapter
60054	EN	Average time to retrieve shadow table results (ms)	milliseconds	If this value is increasing over time, it indicates that the source database may be slow to respond.	DB2LogReaderAdapter
60055	EN	Total rowsets sent out	count	A row set can have multiple rows up to max batch size.	DB2LogReaderAdapter
60056	EN	Total rows sent out	count	Total number of rows sent out.	DB2LogReaderAdapter
60057	EN	Average rows per rowset	count	Average number of rows per rowset: (Total count of rows sent out/ Total count of rowsets sent out). When there are many small transactions, this value is closer to 1.0.	DB2LogReaderAdapter
60058	EN	Average time to send rowsets (ms)	milliseconds	Average time to send rowsets to DPServer, in milliseconds.	DB2LogReaderAdapter
60059	EN	Latest timestamp applier sent rows	timestamp	Most recent time the adapter's applier sent rows to DPServer.	DB2LogReaderAdapter
60060	EN	Latest timestamp applier sent rows (UTC)	timestamp	Most recent time the adapter's applier sent rows to DPServer (UTC).	DB2LogReaderAdapter
60061	EN	Latest applier sent sequence ID	timestamp	Sequence ID of the last row the adapter's applier has sent to DPServer.	DB2LogReaderAdapter
60062	EN	Latest received truncated sequence ID	timestamp	Sequence ID of the last row received by the DPServer.	DB2LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
60063	EN	Total scans	count	Cumulative number of total scans, starting from the most recent start or resume of the adapter instance. This number should be increasing if any of the remote source tables have continuous changes.	DB2LogReaderAdapter
60064	EN	Max scan time per record over the latest statistics update interval (ms)	milliseconds	If this value is larger than 0 seconds, the source database may be busy.	DB2LogReaderAdapter
60065	EN	Trigger queue records in the slowest scan	count	Number of trigger queue records in the slowest scan over the latest statistics update interval (MAX_SCAN_TIME_PER_RECORD equals the value above).	DB2LogReaderAdapter
60066	EN	Min scan time per record over latest statistics update interval (ms)	milliseconds	If this value is larger than 0 seconds, the source database may be busy.	DB2LogReaderAdapter
60067	EN	Trigger queue records in the fastest scan	count	Number of trigger queue records in the fastest scan over the latest statistics update interval (MIN_SCAN_TIME_PER_RECORD equals the value above).	DB2LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
60068	EN	Records in shadow table	count	Value string: <number of records in shadow table>;<remotetableID>;"<shadow table schema>". "<shadow table name>"	DB2LogReaderAdapter
60069	EN	Average transaction size	count	Overall average transaction size since the remote source started.	DB2LogReaderAdapter
60070	EN	Dispersion of transaction size	count	Standard variance of transaction size since the remote source started.	DB2LogReaderAdapter
70001	EN	Time statistics obtained	timestamp	Day, date, and time when statistics were obtained	DB2Mainframe-LogReaderAdapter
70002	EN	Time replication last started	timestamp	Day, date, and time that Replicating state started	DB2Mainframe-LogReaderAdapter
70003	EN	Time statistics last reset	timestamp	Day, date, and time that statistics counters were reset	DB2Mainframe-LogReaderAdapter
70004	EN	Items held in Global LRUCache	count	Number of object references in the internal Least Recently Used cache	DB2Mainframe-LogReaderAdapter
70005	EN	Number of operations replicated	count	Number of operations replicated, excluding BEGIN/COMMIT/ROLLBACK operations	DB2Mainframe-LogReaderAdapter
70006	EN	Number of transactions replicated	count	Number of transactions replicated	DB2Mainframe-LogReaderAdapter
70007	EN	Average PDB Service Time / Operation	n/a	Average transaction log operation arrival rate: (average XLog access times) / number of operations processed	DB2Mainframe-LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
70008	EN	Operation Queue Size	count	Operation Queue Size	DB2Mainframe-LogReaderAdapter
70009	EN	Operation Data Hash Size	count	Operation Data Hash Size	DB2Mainframe-LogReaderAdapter
70010	EN	Number of transactions truncated	count	Number of transactions truncated	DB2Mainframe-LogReaderAdapter
70011	EN	Current operation queue size	count	Current number of operations in the operation queue. A low value may suggest a performance bottleneck in the operation processor component. A high value may suggest a performance bottleneck in downstream components like sender.	DB2Mainframe-LogReaderAdapter
70012	EN	Log reposition point locator	n/a	Position in the log that the scan will start from	DB2Mainframe-LogReaderAdapter
70013	EN	Log reposition point LSN	n/a	LSN in the log that the scan will start from	DB2Mainframe-LogReaderAdapter
70014	EN	Last processed operation LSN	n/a	LSN of the last processed operation	DB2Mainframe-LogReaderAdapter
70015	EN	Total operation process time (s)	seconds	Time used by the UDB Operation Processor, in seconds	DB2Mainframe-LogReaderAdapter
70016	EN	Average sender operation processing time (ms)	milliseconds	Average time to send an operation, in milliseconds	DB2Mainframe-LogReaderAdapter
70017	EN	Average sender operation wait time (ms)	milliseconds	Average time between sending operations, in milliseconds	DB2Mainframe-LogReaderAdapter
70018	EN	Average row set send time (ms)	milliseconds	Average time to send a row set, in milliseconds	DB2Mainframe-LogReaderAdapter
70019	EN	Number of sender operations processed	count	Number of operations sent	DB2Mainframe-LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
70020	EN	Number of row sets sent	count	Number of row sets sent by sender. Processed change data operations are grouped and sent as row sets.	DB2Mainframe-LogReaderAdapter
70021	EN	Average row sets per second	count	Average number of row sets processed by sender per second. A low value may suggest a performance problem in sender or downstream components.	DB2Mainframe-LogReaderAdapter
70022	EN	Last sequence ID sent	n/a	Sequence ID of the last CDC row sent	DB2Mainframe-LogReaderAdapter
70023	EN	Last transaction ID sent	n/a	Transaction ID of the last row set sent to data provisioning server	DB2Mainframe-LogReaderAdapter
70024	EN	Begin marker received	timestamp	Timestamp when the adapter received the begin marker request from DPServer	DB2Mainframe-LogReaderAdapter
70025	EN	Begin marker applied	timestamp	Timestamp when the begin marker was applied to the source	DB2Mainframe-LogReaderAdapter
70026	EN	Begin marker sent	timestamp	Timestamp when the begin marker was read from the source and sent to DPServer	DB2Mainframe-LogReaderAdapter
70027	EN	End marker received	timestamp	Timestamp when the end marker was received by the adapter from DPServer	DB2Mainframe-LogReaderAdapter
70028	EN	End marker applied	timestamp	Timestamp when the end marker was applied to the source	DB2Mainframe-LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
70029	EN	End marker sent	timestamp	Timestamp when the end marker was read from the source and sent to DPServer	DB2Mainframe-LogReaderAdapter
70030	EN	Average time of JNI call	timestamp	Average time taken to receive rowset from log-reader	DB2Mainframe-LogReaderAdapter
70031	EN	Number of JNI call	count	Number of JNI calls made to the logreader	DB2Mainframe-LogReaderAdapter
70032	EN	Number of Row-sets read from JNI interface	count	Number of rowsets requested from logreader	DB2Mainframe-LogReaderAdapter
70033	EN	Average time for a Rowset to stay in the queue (ms)	milliseconds	Average time a rowset stays in the logreader queue, in milliseconds	DB2Mainframe-LogReaderAdapter
70034	EN	Average Rowsets read per second	count	Average number of rowsets read by logreader per sec-ond.	DB2Mainframe-LogReaderAdapter
70035	EN	Number of reads through JNI inter-face	count	Number of rowsets read through log-reader	DB2Mainframe-LogReaderAdapter
70036	EN	Number of time-out in JNI interface	count	Number of JNI in-terface timeouts in logreader	DB2Mainframe-LogReaderAdapter
70037	EN	Average time to read a Rowset (ms)	milliseconds	Average time log-reader takes to read a rowset from replication agent, in milliseconds	DB2Mainframe-LogReaderAdapter
70038	EN	Average wait time when reading a rowset (ms)	milliseconds	Average time wait-ing for transaction rowset from repli-cation agent, in milliseconds	DB2Mainframe-LogReaderAdapter
70039	EN	Number of Nodes received from Rep-Agent	count	Number of nodes logreader received from Replication Agent	DB2Mainframe-LogReaderAdapter
70040	EN	Number of Nodes released in C++ codes	count	Number of nodes processed and released by log-reader	DB2Mainframe-LogReaderAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
70041	EN	Average number of Nodes processed per second	count	Average number of nodes processed by logreader per second	DB2Mainframe-LogReaderAdapter
70042	EN	Number of commands received from RepAgent	count	Number of commands logreader received from Replication Agent	DB2Mainframe-LogReaderAdapter
70043	EN	Average number of commands processed per second	count	Average number of commands processed by logreader per second	DB2Mainframe-LogReaderAdapter
70044	EN	Begin marker received	timestamp	Timestamp when the adapter received the begin marker request from DPServer.	DB2MainframeAdapter
70045	EN	Begin marker applied	timestamp	Timestamp when the begin marker was applied to the source.	DB2MainframeAdapter
70046	EN	Begin marker sent	timestamp	Timestamp when the begin marker was read from the source and sent to DPServer.	DB2MainframeAdapter
70047	EN	End marker received	timestamp	Timestamp when the end marker was received by the adapter from DPServer.	DB2MainframeAdapter
70048	EN	End marker applied	timestamp	Timestamp when the end marker was applied to the source.	DB2MainframeAdapter
70049	EN	End marker sent	timestamp	Timestamp when the end marker was read from the source and sent to DPServer.	DB2MainframeAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
70050	EN	Invalid scans	count	Cumulative number of invalid scans, starting from the most recent start or resume of the adapter instance. If this number is increasing, there are no changes to any of the source tables of a remote source.	DB2MainframeAdapter
70051	EN	Average time to scan the trigger queue (ms)	milliseconds	Average time taken to scan the trigger queue, in milliseconds, calculated from the most recent start or resume of the adapter instance. This value is calculated for the active processing time, and does not include time when the adapter instance was idle. (Total cumulative scan time/Total scans)	DB2MainframeAdapter
70052	EN	Total trigger queue records scanned	count	Total number of records scanned by adapter for a remote source.	DB2MainframeAdapter
70053	EN	Total scan batches	count	In each scan, trigger queue records are split into batches based on table name and max batch size	DB2MainframeAdapter
70054	EN	Average trigger queue records per scan	count	Average number of trigger queue records per scan	DB2MainframeAdapter
70055	EN	Average records per batch	count	Average number of records per batch: (Total number of trigger queue records scanned/Total number of scan batches)	DB2MainframeAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
70056	EN	Average time to scan trigger queue, per record (ms)	milliseconds	Average time taken per record to retrieve the trigger queue and split into different batches, in milliseconds	DB2MainframeAdapter
70057	EN	Total transactions	count	Total number of transactions processed	DB2MainframeAdapter
70058	EN	Unscanned trigger queue records	count	Current number of unscanned trigger queue records. If this value is decreasing, replication is catching up with the source changes. Otherwise replication may lag	DB2MainframeAdapter
70059	EN	Latest scanned trans_time in trigger queue	timestamp	Latest scanned trans_time in the trigger queue. If this value is closer to 'Latest trans_time in trigger queue' then there is no lag in scanning.	DB2MainframeAdapter
70060	EN	Latest trans_time in trigger queue	timestamp	Latest trans_time (i.e. max) in the trigger queue. If this value changes continuously then there are changes to source tables in this remote source	DB2MainframeAdapter
70061	EN	Max scan number in trigger queue	number	Max scan number in trigger queue	DB2MainframeAdapter
70062	EN	Times shadow table results retrieved	count	Number of times the shadow table results have been retrieved	DB2MainframeAdapter
70063	EN	Average time to retrieve shadow table results (ms)	milliseconds	If this value is increasing over time, it indicates that the source database may be slow to respond.	DB2MainframeAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
70064	EN	Total rowsets sent out	count	A row set can have multiple rows up to max batch size	DB2MainframeAdapter
70065	EN	Total rows sent out	count	Total number of rows sent out	DB2MainframeAdapter
70066	EN	Average rows per rowset	count	Average number of rows per rowset: (Total count of rows sent out/ Total count of rowsets sent out). When there are many small transactions, this value is closer to 1.0	DB2MainframeAdapter
70067	EN	Average time to send rowsets (ms)	milliseconds	Average time to send rowsets to DPServer, in milliseconds	DB2MainframeAdapter
70068	EN	Latest timestamp applier sent rows	timestamp	Most recent time the adapter's applier sent rows to DPServer	DB2MainframeAdapter
70069	EN	Latest timestamp applier sent rows (UTC)	timestamp	Most recent time the adapter's applier sent rows to DPServer (UTC)	DB2MainframeAdapter
70070	EN	Latest applier sent sequence ID	timestamp	Sequence ID of the last row the adapter's applier has sent to DPServer	DB2MainframeAdapter
70071	EN	Latest received truncated sequence ID	timestamp	Sequence ID of the last row received by the DPServer	DB2MainframeAdapter
70072	EN	Total scans	count	Cumulative number of total scans, starting from the most recent start or resume of the adapter instance. This number should be increasing if any of the remote source tables have continuous changes.	DB2MainframeAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
70073	EN	Max scan time per record over the latest statistics update interval (ms)	milliseconds	If this value is larger than 0 seconds, the source database may be busy.	DB2MainframeAdapter
70074	EN	Trigger queue records in the slowest scan	count	Number of trigger queue records in the slowest scan over the latest statistics update interval (MAX_SCAN_TIME_PER_RECORD equals the value above)	DB2MainframeAdapter
70075	EN	Min scan time per record over latest statistics update interval (ms)	milliseconds	If this value is larger than 0 seconds, the source database may be busy.	DB2MainframeAdapter
70076	EN	Trigger queue records in the fastest scan	count	Number of trigger queue records in the fastest scan over the latest statistics update interval (MIN_SCAN_TIME_PER_RECORD equals the value above)	DB2MainframeAdapter
70077	EN	Records in shadow table	count	Value string: <number of records in shadow table>;<remotetableID>;"<shadow table schema>". "<shadow table name>"	DB2MainframeAdapter
70078	EN	Average transaction size	count	Overall average transaction size since the remote source started.	DB2MainframeAdapter
70079	EN	Dispersion of transaction size	count	Standard variance of transaction size since the remote source started.	DB2MainframeAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
80001	EN	Read delta link count	count	Total number of delta links read and processed, starting from the most recent start or resumption of the adapter instance. Each delta link contains a set of change records.	CloudDataIntegrationAdapter
80002	EN	Read link count	count	Total number of all delta links and next links processed by the adapter, starting from the most recent start or resumption of the adapter instance.	CloudDataIntegrationAdapter
80003	EN	Average time waiting for links (ms)	milliseconds	Average time spent waiting for the CDI provider to respond to adapter requests to fetch data from delta or next links, in milliseconds.	CloudDataIntegrationAdapter
80004	EN	Average time reading links (ms)	milliseconds	Average time spent reading and parsing data from delta or next links, in milliseconds.	CloudDataIntegrationAdapter
80005	EN	Average time processing links (ms)	milliseconds	Average time spent converting parsed data into rowsets, in milliseconds.	CloudDataIntegrationAdapter
80006	EN	Number of records read	count	Total number of records read, starting from the most recent start or resumption of the adapter instance.	CloudDataIntegrationAdapter
80007	EN	Start time of first delta link	timestamp	Timestamp when the adapter read the first delta link from the CDI provider.	CloudDataIntegrationAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
80008	EN	End time of first delta link	timestamp	Timestamp when the adapter finished reading, parsing, and converting rowsets for the first delta link.	CloudDataIntegrationAdapter
80009	EN	Records in first delta link	count	Number of records read from the first delta link.	CloudDataIntegrationAdapter
80010	EN	Start time of latest delta link	timestamp	Timestamp when the adapter fetched the most recent delta link from the CDI provider.	CloudDataIntegrationAdapter
80011	EN	End time of latest delta link	timestamp	Timestamp when the adapter finished reading, parsing, and converting rowsets for the most recent delta link.	CloudDataIntegrationAdapter
80012	EN	Records in latest delta link	count	Number of records read from the most recent delta link.	CloudDataIntegrationAdapter
80013	EN	Begin marker received	timestamp	Timestamp when the adapter received the begin marker request from DPServer.	CloudDataIntegrationAdapter
80014	EN	Begin marker sent	timestamp	Timestamp when the begin marker was read from the source and sent to DPServer.	CloudDataIntegrationAdapter
80015	EN	End marker received	timestamp	Timestamp when the end marker was received by the adapter from DPServer.	CloudDataIntegrationAdapter
80016	EN	End marker sent	timestamp	Timestamp when the end marker was read from the source and sent to DPServer.	CloudDataIntegrationAdapter
90010	EN	Latest pointer	ODP pointer string	The pointer of the latest delta polled on the extractor.	ABAPAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
90020	EN	Sent rowset count	count	The number of the rowsets sent for the extractor.	ABAPAdapter
90021	EN	Average number of records in a rowset	count	Average number of records in a rowset.	ABAPAdapter
90022	EN	Average time to send a rowset	milliseconds	Average time to send a rowset.	ABAPAdapter
90023	EN	Latest sent time	timestamp	The timestamp of the most recent sending of a rowset for the extractor.	ABAPAdapter
90025	EN	Latest sent sequence id	timestamp	The sequence id sent most recently for the extractor.	ABAPAdapter
90042	EN	Upsert count	count	Count of UPSERT records.	ABAPAdapter
90043	EN	Insert count	count	Count of INSERT records.	ABAPAdapter
90044	EN	Delete count	count	Count of DELETE records.	ABAPAdapter
90050	EN	Open calls count	count	The number of ODP OPEN calls (deltas polled) for the extractor.	ABAPAdapter
90051	EN	Average open call time	milliseconds	The average time an ODP OPEN call took for the extractor	ABAPAdapter
90052	EN	Average number of records in a delta	count	Average number of records in a delta for the extractor.	ABAPAdapter
90053	EN	Fetch call count	count	The number of ODP FETCH calls for the extractor.	ABAPAdapter
90054	EN	Average fetch call time	milliseconds	The average time an ODP FETCH call took for the extractor.	ABAPAdapter
90055	EN	Max fetch time per record	milliseconds	Maximum fetch time per record for the extractor.	ABAPAdapter
90056	EN	Record count in the slowest fetch	count	Record count in the slowest fetch.	ABAPAdapter

ID	Language	Statistic Name	Unit	Long Description	Component
90057	EN	Min fetch time per record	milliseconds	Minimum fetch time per record.	ABAPAdapter
90058	EN	Record count in the fastest fetch	count	Record count in the fastest fetch.	ABAPAdapter

3.11 Configure User Settings Profiles

By creating user settings profiles, you can quickly switch between different monitor layouts. Settings profiles contain information about visible columns, column order, column width, column filters, table visibility, and slider positions.

Context

You can create, modify, or remove settings profiles in each Data Provisioning Monitor by clicking the *Settings* button.

Procedure

- To add a new settings profile, click *Add*.
  a. Specify a name for the profile and whether to make it the default profile.
  b. Click *Add*.

  A new profile is created using the current layout and column display settings.
- To switch to an existing settings profile, select the profile and click *Load*.
  The current layout and column display settings are updated from the settings saved in the profile.
- To modify an existing settings profile, select the profile and click *Update*.
  a. If you want to make the profile the default profile, select *Default*.
  b. Click *Update*.

  The selected profile is updated with the current layout and column display settings.
- To remove an existing settings profile, select the profile and click *Delete*.
  The selected profile is removed from the *Profiles* table.

3.12 Create Notifications

Create e-mail notifications for various task, remote subscription, and design-time object statuses.

Prerequisites

The user must have the following roles or privileges to create status notifications:

Table 24: Roles and Privileges

Action	Role or Privilege
Enable users to schedule task	Role: sap.hana.xs.admin.roles::JobSchedulerAdministrator
Configure SMTP	Role: sap.hana.xs.admin.roles::SMTPDestAdministrator
Create status notifications	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::NotificationAdministration

To activate status notifications, the following must occur:

- Scheduling must be enabled in the XS Job Admin Dashboard at `http://<host>:<port>/sap/hana/xs/admin/jobs`.
  The user that enables other users to schedule needs the role *sap.hana.xs.admin.roles::JobSchedulerAdministrator*.
- To create notifications, the job *sap.hana.im.dp.monitor.jobs/checkNotifications* must be enabled in the *XS Job Details* page: `http://<host>:<port>/sap/hana/xs/admin/jobs/#/package/sap.hana.im.dp.monitor.jobs/job/checkNotifications`.
- A sender address must be configured in the monitor. If no sender address is configured, notification emails won't be sent.
  To configure the sender address, click *Settings* > *Set sender email address* in the monitor.
- The SAP HANA SMTP mail client must be correctly configured in the *SMTP Configurations* page at `http://<host>:<port>/sap/hana/xs/admin/index.html#smtp`.
  The user who configures the SMTP client needs the role *sap.hana.xs.admin.roles::SMTPDestAdministrator*

Context

You can create notifications for the following statuses:

Object Type	Supported Statuses
Task execution	COMPLETED
	FAILED
	CANCELLED

Object Type	Supported Statuses
Remote subscription	ERROR
	WARNING
Design time object	COMPLETED
	FAILED
	CANCELLED
	ERROR
	WARNING

Procedure

1. In the *Task Overview*, *Remote Subscription Monitor*, or *Design Time Objects* table, select the object for which you want to create a notification.
2. Click the *Notifications* button.
   The list of notifications for the object is displayed.
3. Click the *Add* button to create a new notification.
4. Specify a name, status conditions, and recipient e-mail addresses for the notification.
5. If you want to enable the notification immediately, select *Is active*.
6. Click *Create Notification*.
   The new notification is added to the list of notifications for the object. When the conditions for the notification are met, users in the recipient list are sent an e-mail containing details about the event that triggered the notification.

Related Information

Assign Roles and Privileges

3.13 Creating Monitoring Alerts

Create monitoring alerts for various functions.

You can receive alerts for the following functions:

- Agent availability
- Agent memory usage
- Remote subscription exception

- Data Quality reference data
- Long-running tasks

You can configure monitoring alerts from the SAP HANA cockpit. For more information, see the *SAP HANA Administration Guide*.

Related Information

Monitoring Alerts (SAP HANA Administration Guide)
SAP HANA Administration Guide: Configuring Alerts

4 Creating and Monitoring Data Assurance Jobs

Use the Data Assurance Job Monitor to create Data Assurance jobs to monitor data quality and integrity when replicating data into SAP HANA. The Job Monitor also lets you drop, alter, abort, and schedule jobs, as well as view reports, manage job history, and create alert notifications.

The Data Assurance Monitoring UI is available on SAP HANA XS Classic.

> ⓘ Note
>
> Create only one job per remote subscription, and only remote subscriptions with a replication behavior of *Initial + realtime* and *Realtime only* are supported. If you use a remote subscription with other replication behaviors, the Data Assurance Monitor UI generates an error.

Initialization

Before creating Data Assurance jobs and other functions, you must initialize to take advantage of the underlying APIs created when you configured your remote source. This initialization takes place when you choose your schema and remote source. By default, the Job monitor is set to the `SAP_HANA_IM_DP` schema. You can choose others.

Once this initialization has occurred, you're now able to use the Job Monitor.

The Data Assurance Job Monitor User Interface

The Job Monitor user interface is populated with buttons to make creating, editing, and deleting a job easy. Through the UI, you can also configure trace logging, history reports, notifications, and so on.

Creating a Job

When you click the *Create Job* button, you're asked to select a remote subscription from the remote source. You then configure the necessary job options. You can run a job by configuring the *Compare mode* option, but if you want to configure other options, click *Show Advanced Options*.

Dropping a Job

You can drop a job by selecting a job on the Job Monitor and clicking the *Drop options* menu button ("trashcan"). Here you're also given an option to "cascade" your drop, meaning that you want to drop all of the resources this job relies on (not including connections).

Data Assurance Settings

Menu Item	Description
Compress Logs	Compresses any logs older than the given timeframe when this action is performed. After logs are compressed, they can't be seen and accessed from the UI. For example, a report wouldn't be available.
Uncompress Logs	If you want to view and access your already compressed logs, uncompress them using this functionality.
Purge Logs	Deletes log files after the specified time.
Purge History	Deletes history files.
Trace	Trace settings are populated with default levels. We recommend that you keep the default settings. Change these default levels only after consulting with SAP support. Trace files are located in the `<DPAgent_root>/log` directory, and have a naming convention of `da_0.log`, `da_1.log`, and so on.
Configuration	Configure Data Assurance job parameters for better performance and efficiency.
Job Option Constraints	Displays the default values, value ranges, and other information about Data Assurance job options.

Related Information

Data Assurance Job Options [page 125]
Data Assurance Job Monitor Statistics [page 130]
Data Assurance Framework Configuration [page 131]
SAP HANA (Remote) to SAP HANA (Target) Data Type Mapping With Data Assurance Adapter [page 133]
Comparison and Reconciliation Strategies [page 134]
Data Assurance Objects Created Upon Initialization [page 136]
Creating a Data Assurance Subscription that Contains a Computed Column Expression [page 138]
Data Assurance Job Monitor Statistics [page 130]
Data Assurance

4.1 Data Assurance Job Options

Configure your Data Assurance job.

Some options have limits to their value ranges and so on. You can view these limits by navigating to *Data Assurance Setttings* > *Job Option Constraints*.

Option	Description
Compare mode	Specifies the row comparison mode. • *ROW_COMPARE*: Compares all table rows • *KEY_COMPARE*: Compares only the table columns that have been marked as key columns in the compare set • *ROW_COUNT*: Compares the table row counts only • *DIRECT_RECON*: Compares and simultaneously reconciles all rows. • *DIRECT_MAT*: Direct materialization skips comparison and copies the source rows into the target table. **Note** The target table must be empty. Default value: Row compare
Abort if row count mismatch	Determines whether to abort the comparison job if the total row counts for source and target do not match. Valid values: true or false. Default value: false. **Note** This option must be set to false when *Compare mode* is *DIRECT_RECON*.
Maximum row difference count	If row difference count exceeds this number, then the comparison job fails. The number 0 means unlimited. Valid values: 0–9223372036854775807. Default value: 0 if the COMPARE_MODE is direct_recon; 1000 otherwise. **Note** This option must be set to 0 if the *Compare mode* is *DIRECT_RECON*.

Option	Description
Create column log	Generates a column differences log, which lists all missing, orphaned, and inconsistent row values (keys and columns). Create a column log if you want to: • Generate a reconciliation script • Perform nondirect automatic reconciliation • Generate a detailed report Valid values: true or false. Default value: false. **Note** This option must be set to false when *Compare mode* is *DIRECT_RECON*.
Create reconciliation script	Generates a reconciliation script. To use this parameter, you must also set *Create column log* to *true*. Valid values: true or false. Default value: false. **Note** This option must be set to false when *Compare mode* is *DIRECT_RECON*.
Enable row count	Determines whether or not to count source and target table rows before they're compared. The Data Assurance server uses the row count to estimate the comparison progress and end time. **Note** The Data Assurance server counts rows if *Compare mode* is set to *ROW_COUNT*. Use *Enable row count* only if *Compare mode* is a value other than *Row count*. Valid values: true or false Default value: true
Hash type	Specifies the hash type for the comparison. • DATABASE_HASH: Uses the hash function provided by the database. • AGENT_HASH: Uses the hash function specified in the *Column compare mode* option. **Note** If the *Compare mode* parameter is set to *DIRECT_RECON*, *Hash type* must be set to *AGENT_HASH*.

Option	Description
Column compare mode	Specifies how each column is compared. • *Column Hash*: Compares using column's hash value. Each column value gets its own hash. • *Row Hash*: Hashes multiple column values into a single hash. **Note** If the *Hash type* parameter is set to *DATABASE_HASH*, the *Column compare mode* parameter must be set to *Row Hash*. • *Literal* (default): Compares the full column data (value-to-value). **Note** All hash types must be set to *Literal* when *Compare mode* is set to *DIRECT_RECON*.
Number of partitions	Specifies the number of partitions for a table
Maximum concurrent comparisons	The maximum number of comparisons that can run concurrently. This option applies to the owning job only. Valid values: 1 to 231-1 Default value: 5
Enable statistics	Enables the generation of statistics (currently only applies to row-sizing). Valid values: true or false. Default value: false The statistics bytesRead and queueBytes are enabled only when *Enable statistics* is set to *true*.
Compare fields to include	Columns to use for comparison. These columns can be different from the columns used in the subscription SQL statement.
Compare fields to exclude	Columns to exclude during comparison.
Source filter	Filter condition for fetching data from a source
Target filter	Filter condition for fetching data from a target
Pushdown computed column for target	Push down computed columns in the subscription SQL to the target where clause.
Ignore Trailing Spaces	Specifies whether all trailing whitespace characters (0x20) are ignored when comparing/hashing CHAR, NCHAR, VARCHAR, and NVARCHAR data types. If true, trailing spaces are ignored. If false, trailing spaces are counted. Valid values: true or false. Default value: true.

Option	Description
Ignore Zero Padding	If true, padding is ignored. If false, padding is counted. Valid values: true or false. Default value: true.
Ignore Case	Specifies whether case should be ignored when comparing/hashing CHAR, NCHAR, VARCHAR and NVARCHAR data types for non-primary key values.
Scale	Specifies the scale (rounding) used when comparing floating-point values This enables (14.002==14.0017) to be true when *default_comparison_scale* = 3. Valid values: -231 to 231-1 Default value: 10 **Note** This option overrides the framework configuration parameter, *default_comparison_scale*. Valid values: true or false. Default value: false.
Thread count for auto reconcile	An integer value that governs how many direct reconciliation apply threads are employed per Data Assurance compare partition. Increasing this value can improve direct reconciliation performance. Default value: 1. **Note** These apply threads are in addition to the threads created by these configurations: Framework configuration option: *comparer_max_concurrent_threads* Job option: *Maximum concurrent comparisons*
Source fetch size	Standard JDBC enables you to specify the number of rows fetched with each source database round trip for a query, and this number is referred to as the fetch size.
Target fetch size	Standard JDBC enables you to specify the number of rows fetched with each target database round trip for a query, and this number is referred to as the fetch size.
Timestamp format	Specifies the TIMESTAMP format to use to parse strings that are to be compared to (JDBC) DATE. Default value: yyyy-MM-dd HH:mm:ss.SSS

Option	Description
Date format	Specifies the DATE format to use to parse strdaings that are to be compared to (JDBC) DATE. Default value: yyyy-MM-dd
Time format	Specifies the TIME format to use to parse strings that are to be compared to (JDBC) DATE. Default value: HH:mm:ss.SSS
Retry differences	Specifies the retry option. • *Never*: No recompare • *Wait and retry*: Run the re-compare based on the *Maximum retries* and *Retry delay in seconds* options. Default value: never. **Note** This option must be set to *Never* when *Compare mode* is set to *DIRECT_RECON*. **Note** "Wait and retry" recompares only those rows where a difference was found in the initial compare. For example, if a compare finds 10K differences, when you retry, it does a compare only on those 10K rows.
Maximum retries	Specifies the total number of recomparison for rows that have differences resulting from a previous comparison. Valid values: 0–100. Default value: 3.
Retry delay in seconds	Specifies the number of seconds delay for each recomparison. Valid values: 0–86400. Default value: 10.
Immediately abort duplicate key	Specifies whether the comparison aborts immediately on duplicate key detection. • *true*: Abort immediately on duplicate key detection. • *false*: The comparison aborts after the current comparison is complete Valid values: true or false. Default value: false.

Option	Description
Automatically reconcile	Indicates whether to automatically apply the reconciliation script. If you want to reconcile data using database hash during the comparison process, set *Compare Mode* to *ROW_COMPARE*, *Hash type* to *DATABASE_HASH*, *Create Column Log* to *true*, and *Automatically reconcile* to *true*. This option is automatically set to *true* when *Compare Mode* is set to *DIRECT_RECON* or *DIRECT_MAT*. Valid values: true or false. Default value: false.

4.2 Data Assurance Job Monitor Statistics

Keep track of your Data Assurance job with Data Assurance Job Monitor.

The Data Assurance Job Monitor provides helpful information about the status of your Data Assurance job and gives you insight into any potential issues with the completeness and integrity of your data.

Statistic	Description
Job Name	Specifies the name of your Data Assurance job
Job Type	The value that appears here what you chose for the Compare Mode parameter when creating this Data Assurance job.

Statistic	Description
Status	The following values may appear. • SUCCESS: The job has executed successfully. • ABORTED: The job has been aborted. This usually means that you terminated a running job. • CANCELED: The job has been cancelled. This usually means that you terminated the job while is was still pending. For example, the job could have been waiting for an open connection or thread when you stopped it. • FAILED: The job has failed. • FAILED_NOT_FOUND: The job is not found. • ARCHIVED: The job has been archived to history. • RUNNING: The job is currently running • WARNING: The API executed, but with a warning • WAITING_RETRY_NO_THREAD: The job is waiting to retry because there is no available thread. • WAITING_RETRY_WITH_THREAD: The job is waiting to retry and there is a thread available. • PENDING_NOT_SUBMITTED: The job was already submitted to execute, but it is in a pending status, and it has not been submitted because of a resource limitation. • PENDING_SUBMITTED: This is the initial status when a job has just been submitted to execute. • NOT_EXECUTED: The job has not be submitted for execution.
Message	The message will return an error for each API, if an error occurs.
Start Time	Specifies the start time of your Data Assurance job.
End Time	Specifies the end time of your Data Assurance job.
Progress	Specifies the progress of the job, from 0-100%
Rows Read	Specifies the number of rows read during the run of the Data Assurance job.
Missing	Specifies the number of rows in the primary table that are not present in the replicated table.
Orpahn	Specifies the number of rows in the replicated table that are not present in the primary table.
Inconsistent	Specifies the number of rows that are present in both the replicated and primary tables, but the column data is different.
Reconciled	Specifies the number of rows that were reconciled.
Report	After a job is complete, links to a report about the job appear here. Reports are available in TEXT and XML formats.

4.3 Data Assurance Framework Configuration

Configure Data Assurance job parameters for better performance and efficiency.

You can find the configuration parameters in common menu under *Data Assurance Settings* > *Configuration*.

Option	Description
auto_recon_stmt_batch_size	Specifies the number of SQL statements the Data Assurance Server sends to the Data Assurance Agent in one (remote) invocation.
boundaries_stored	Specifies the number of boundary keys to save per compare set for future partition generation
boundary_sample_size	Specifies the number of boundary keys to sample and keep in memory during a comparison
boundary_sample_step	Specifies the number of rows skipped between each boundary key sampling
comparer_max_concurrent_threads	Specifies the maximum concurrent threads the comparer uses when comparing (maximum concurrent comparisons).
comparer_predicate_range_max_cols_in_sql	Specifies the maximum number of columns Data Assurance uses when generating predicate WHERE clauses.
comparer_predicate_range_max_varying_cols_in_sql	Specifies the maximum number of columns that vary in value, that Data Assurance uses when generating predicate WHERE clauses.
comparer_recently_finished_ttl_secs	Specifies the number of seconds for which run information of a completed job remains in the 'recently finished' view.
comparer_retry_delay_threshold_secs	Specifies the (maximum) number of seconds a comparison holds on to a compare thread ready for a retry.
comparer_retry_max_keys_per_clause	Specifies the maximum number of "single" keys in a WHERE clause.
comparer_retry_min_fill_percent	Specifies the fill percent required before 'single' keys are combined into ranges.
comparer_retry_min_fill_percent_literal	Specifies the fill percent required before 'single' keys are combined into ranges, for LITERAL comparisons.
comparer_retry_min_keys_in_range	Specifies the minimum number of keys to use before calculating key ranges.

comparer_retry_strategy	The strategy for selecting rows:
	`ALL_RANGE, // Single range: WHERE col_id >= first_Different_Primary_Key AND col_id <= last_Different_Primary_Key`
	`IN_BATCHES, // For single-col Primary_Key: WHERE IN(...); For multi-col Primary_Keys: WHERE ((PK1=V1 AND PK2=V2) OR (PK1=...) OR (PK1=...))`
	`DEFAULT // Wraps high-density Primary_Keys into range and leaves low-density Primary_Keys as WHERE col_id IN (...)`

db_connection_pool_idle_timeout_secs	Specifies the number of seconds an open connection can remain idle before it's closed.
db_connection_pool_size	Specifies the maximum number of connections that are made available in the connection pool (per connection profile).
db_connection_retry_interval	Specifies the number of seconds to wait between connection attempts.
db_connection_retry_times	Specifies how many times the connection manager must attempt to connect to the database.
db_hash_hana_ignore_null	Specifies whether to ignore a null value when calculating a multihash value for HANA.
default_column_compare_mode	Specifies the default column compare mode.

Option	Description
default_comparison_abort_diff_max	Specifies the default maximum number of differences Data Assurance detects before automatically aborting a comparison. A value of zero disables this feature.
default_comparison_scale	Specifies the maximum number of scale digits (that is, digits to the right of a decimal point) that are compared by default. If the number of digits in the scale exceeds this number, the scale is rounded half-up. A negative value disables this feature.
direct_recon_target_queue_size	Specifies the maximum number of missing, orphaned, and inconsistent rows to store in the (blocking) queue. This queue is shared by all reconciliation apply threads.
enable_report_generator	When TRUE, report files are auto-generated (as necessary) when the job history is viewed.
file_output_encoding	Specifies the file output encoding for all reconciliation scripts and report files.
lob_fetch_size	Specifies the maximum number of Large Object (LOB) bytes to read and compare.
recon_tran_max_stmts	Specifies the maximum number of statements per reconciliation transaction. If the number of statements required is greater than this number, multiple transactions must be employed. A value of zero means unlimited.
text_report_diff_page_size	Specifies the maximum number of rows to show in each 'page' of differences.
text_report_max_column_width	Specifies the maximum width of a column in a text report.
text_report_max_line_length	Specifies the maximum length of a line in a text report.

4.4 SAP HANA (Remote) to SAP HANA (Target) Data Type Mapping With Data Assurance Adapter

The following table shows the data type conversion between a remote SAP HANA source and an SAP HANA target when using the Data Assurance Adapter.

Remote SAP HANA Data Type	Target SAP HANA Data Type
ALPHANUM	NVARCHAR
BIGINT	BIGINT
BINARY	BINARY
BINTEXT	NCLOB
BLOB	BLOB
BOOLEAN	VARCHAR
CHAR	CHAR

Remote SAP HANA Data Type	Target SAP HANA Data Type
CLOB	CLOB
DECIMAL	DECIMAL
DOUBLE	DOUBLE
DATE	DATE
INTEGER	INTEGER
NCHAR	NCHAR
NCLOB	NCLOB
NVARCHAR	NVARCHAR
REAL	REAL/FLOAT
SECONDDATE	TIMESTAMP
SHORTTEXT	NVARCHAR
SMALLDECIMAL	DECIMAL
SMALLINT	INTEGER
TEXT	NCLOB
TIME	TIME
TIMESTAMP	TIMESTAMP
TINYINT	INTEGER
VARBINARY	VARBINARY
VARCHAR	VARCHAR

4.5 Comparison and Reconciliation Strategies

Non-direct Reconciliation

With non-direct reconciliation, two or more comparison phases must complete before any reconciliation phase can begin.

The non-direct reconciliation phases in Data Assurance include:

- Initial comparison
- In-flight data option

- Verify differences
- Reconciliation

Initial Comparison

During an initial comparison, which is mandatory for all jobs, the agent fetches rows from source and target databases using a query. You can specify row comparison when you create a Data Assurance job by specifying any of these options:

- Column hash (`column_hash`) – each column value gets its own hash.
- Row hash (`row_hash`) – hashes multiple column values into a single hash.
- Literal compare – compares the full column data (value-to-value).

> **Note**
>
> Some, such as `column_hash` and `row_hash`, apply only to row comparison jobs.

In-Flight Data Option

Row differences may arise during comparison, due to data being in flight during replication. Data Assurance lets you recheck row differences, by selecting row data only from the target database; you need not run a full table check.

Row differences are classified into three types:

- Missing – a row in the primary table is not present in the replicate table.
- Orphaned – a row in the replicate table is not present in the primary table.
- Inconsistent – a row is present in both tables, but the column data is different.

If Data Assurance identified row differences in the initial comparison, an in-flight data comparison rechecks those rows to verify whether the differences have been reconciled. This is important, especially in replication environments where there are time lags in updating target databases.

In-flight data comparisons, which are optional, apply a “wait and retry” technique to any number of rows that shows data discrepancy during the initial comparison. For example, if an initial comparison at 8:00 p.m. reveals an out-of-sync row, and the wait period is 120 seconds, the recomparison is not started until 8.02 p.m to allow replication to apply any in-flight changes to that row.

> **Note**
>
> In-flight comparisons do not impact the source database. All source rows which differed are cached for recomparison against rows that are reselected from the target database.

Verify Differences

Data Assurance fetches the literal data of all rows that differ between the source and target databases, and writes it to a column log. When you create a job, enable this option by setting `Create column log` to true. A column log lists all the missing, orphaned, and inconsistent row values (keys and columns).

Reconciliation

Based on your job settings, you can reconcile the data differences—either automatically or by generating a reconciliation script. Data Assurance verifies the differences and generates a SQL statement that ensures the target table is in the same state as the source table. Based on the row difference type, Data Assurance runs:

- `insert` statements on the target table for missing rows.

- `delete` statements on the target table for orphaned rows.
- `update` statements on the target table for inconsistent rows.

Direct Reconciliation

With direct reconciliation, compare and reconcile occurs simultaneously in a single phase.

Direct reconciliation implements faster reconciliation for these scenarios:

- Materialization – populate or repopulate the target table using all source rows.
- Reconcile the target table rows without replication latency. All differences presumed to be unchanging need to be reconciled as soon as possible.

To enable faster reconciliation, direct reconciliation:

- Compares and reconciles rows in a single step, in memory.
- Applies the appropriate `insert`, `delete`, or `update` statement directly to rows in the target table that are identified as different from the source table.
- Does not save row values to disk.
- Does not produce a detailed text or XML report (though there are still counters in the job history).
- Enables multi-threaded reconciliation per comparison partition.

4.6 Data Assurance Objects Created Upon Initialization

When you initialize the Data Assurance view in the Data Assurance Monitoring UI, system objects are created on your Data Assurance remote source system.

These system objects (virtual procedures and virtual tables) are the underlying APIs used by the Data Assurance Monitoring UI.

By default, SAP_HANA_IM_DP is the schema where these APIs are generated. However, you can change the schema in the Montoring UI.

Virtual Procedures

Virtual Procedure	Description
AbortJob	Aborts a job
AlterSdiJob	Alters a job's options
CompressLogs	Compresses a job's logs, reports, and SQL files that are older than the time in a given unit
Config	Configures Data Assurance configuration parameters

Virtual Procedure	Description
CreateAndRunJob	Creates a new job and runs it
CreateConnection	Creates a database connection profile, which includes the connectivity parameters in JSON format.
CreateJob	Creates a new job from source/ target objects and options
CreateSdiJob	Creates a new job from source/ target objects and options for the subscription
DropConnection	Drops a connection
DropHistory	Drops all job history for a job
DropJob	Drops a job
Help	Prints help for virtual procedures and views
Initialize	Initializes virtual procedures for Data Assurance
IsSubscriptionSupported	Queries whether given remote subscription was supported in the Data Assurance adapter
JobHistory	Shows job history for a specific job
MonitorJob	Shows runtime status information about running jobs or reports for the jobs that have finished.
PurgeHistory	Truncates job history that older than the time in a given unit.
PurgeLogs	Purges files generated during a compare, such as reports, SQL files, and so on, that are older than the specified time.
RunJob	Runs a job immediately and returns when finished, or waits for the specified number of seconds.
ShowReport	Shows a job report for a special job history ID
Trace	Changes the trace level
UnCompressLogs	Uncompresses job logs, reports, and SQL files that are older than the specified time.

Virtual Tables

View	Description
Boundaries	Shows all of the boundaries
Configurations	Shows Data Assurance configuration parameters
Connections	Shows the connection profile to the database
Jobs	Shows all of the jobs
OptionConstraints	Shows all of the default value and value ranges for comparison option parameters
Traces	Shows all of the trace levels

4.7 Creating a Data Assurance Subscription that Contains a Computed Column Expression

Create or Alter a subscription that contains a computed column.

Context

Procedure

1. Create a job with a computed column in the subscription.

```
create remote subscription "SUB_HANA_LOB_TEST_TAB_H" as
(select PKEY, INT_COL+1 as INT_COL1, LOB_COL, ("VARCHAR_COL" || '_') AS
"VARCHAR_COL"
from "SYSTEM"."V_HANA_LOB_TEST_TAB") target table
"SYSTEM"."T_HANA_LOB_TEST_TAB_H";
-- 1 row compare job with expression
CALL "SYSTEM"."data_assurance_2_CreateSdiJob"(
'SYSTEM','SUB_HANA_LOB_TEST_TAB_H', '{
"compare_mode": "ROW_COMPARE"
}
',?);
```

2. Alter a job with a computed column

```
CALL "SYSTEM"."data_assurance_2_AlterSdiJob"(
'SYSTEM.SUB_HANA_LOB_TEST_TAB_H', '{
    "compare_mode": "ROW_COMPARE",
    "fields":"\"PKEY\",(\"VARCHAR_COL\" || ''_'')  AS \"VARCHAR_COL\"
     }
',?);
```

> **Note**
>
> To provide the double quotes (") and single quotes (') in the fields list with the alterSdiJob API, the double quotes should be escaped as \", and single quotes should be escapted as '', because the monitoring UI will automaticaly do the escaping, so user can directly provide the double and single quotes, such as
>
> ```
> "PKEY","LOB_COL",("VARCHAR_COL" || '_') AS "VARCHAR_COL",("INT_COL" + 1)
> AS "INT_COL1"
> ```

5 Administering Data Provisioning

This section describes common tasks related to the ongoing administration of SAP HANA smart data integration and SAP HANA smart data quality.

Managing Agents and Adapters [page 139]
The Data Provisioning Agent provides secure connectivity between the SAP HANA database and your on-premise sources. Adapters are registered on the agent and manage the connection to your source.

Managing Agent Groups [page 146]
Agent grouping provides failover and load-balancing capabilities by combining individual Data Provisioning Agents installed on separate host systems.

Managing Remote Sources and Subscriptions [page 157]
Remote sources establish the connection between a data provisioning adapter and your source system. Remote subscriptions monitor a remote source for real-time changes to data replicated into the Data Provisioning Server.

Managing Design Time Objects [page 165]
Design time objects such as flowgraphs and replication tasks manage the replication and transformation of data in SAP HANA smart data integration and SAP HANA smart data quality.

Managing Enterprise Semantic Services [page 174]
Use the SAP HANA Enterprise Semantic Services Administration browser-based application to administer and monitor artifacts for semantic services.

Related Information

5.1 Managing Agents and Adapters

The Data Provisioning Agent provides secure connectivity between the SAP HANA database and your on-premise sources. Adapters are registered on the agent and manage the connection to your source.

Manage Agents from the Data Provisioning Agent Monitor [page 140]
Use the agent monitor to perform basic administration tasks such as registering, altering, or dropping Data Provisioning Agents.

Manage Adapters from the Data Provisioning Agent Monitor [page 142]
Use the Data Provisioning Agent Monitor to perform basic administration tasks, such as adding adapters to or removing adapters from a Data Provisioning Agent instance.

Manage Agents Remotely [page 143]
Use the Data Provisioning Agent Management web interface to manage a Data Provisioning agent located within your local network without needing access to the host operating system.

Back Up the Data Provisioning Agent Configuration [page 145]
You can back up your Data Provisioning Agent configuration by copying key static configuration files to a secure location.

Uninstall the Data Provisioning Agent [page 145]
Uninstall the Data Provisioning Agent from a host system using the uninstallation manager.

Parent topic: Administering Data Provisioning [page 139]

Related Information

Managing Agent Groups [page 146]
Managing Remote Sources and Subscriptions [page 157]
Managing Design Time Objects [page 165]
Managing Enterprise Semantic Services [page 174]

5.1.1 Manage Agents from the Data Provisioning Agent Monitor

Use the agent monitor to perform basic administration tasks such as registering, altering, or dropping Data Provisioning Agents.

Prerequisites

The user must have the following roles or privileges to manage agents:

Table 25: Roles and Privileges

Action	Role or Privilege
Add Data Provisioning Agent	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::CreateAgent • System privilege: AGENT ADMIN
Alter Data Provisioning Agent	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::AlterAgent • System privilege: AGENT ADMIN
Remove Data Provisioning Agent	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::DropAgent • System privilege: AGENT ADMIN

Context

Use the following controls in the *Agent Monitor* table to perform an action.

Procedure

- Select *Create Agent* to register a new agent with the SAP HANA system.
  a. Specify the name of the agent and relevant connection information.
  b. If the agent uses a secure SSL connection, select *Enable SSL*.
  c. If you want to assign the agent to an existing agent group, select the group under *Agent Group*.
  d. Click *Create Agent*.

  The new agent appears in the *Agent Monitor* table.
- Select *Alter Agent* to make connection configuration changes on an agent already registered in the SAP HANA system.
  a. Specify the new connection information for the agent. You can't change the name or connection protocol for an existing agent.
  b. If the agent uses a secure SSL connection, check *Enable SSL*.
  c. If you want to assign the agent to an existing agent group, select the group under *Agent Group*.
  d. Click *Alter Agent*.

  The updated agent information appears in the *Agent Monitor* table.
- Select *Drop Agent* to remove an agent from the SAP HANA system.
  a. To drop any dependent objects automatically, such as registered adapters, choose *CASCADE option*. You can't remove an agent while it has dependent objects such as registered adapters. Remove the adapters from the agent manually, or check *CASCADE option*.
  b. Click *Drop Agent*.

  The agent is removed from the *Agent Monitor* table. If the agent was assigned to an agent group, it's also removed from the agent group.

Related Information

ALTER AGENT Statement [Smart Data Integration] [page 265]
CREATE AGENT Statement [Smart Data Integration] [page 276]
CREATE AGENT GROUP Statement [Smart Data Integration] [page 278]
DROP AGENT Statement [Smart Data Integration] [page 291]
DROP AGENT GROUP Statement [Smart Data Integration] [page 293]
Assign Roles and Privileges

5.1.2 Manage Adapters from the Data Provisioning Agent Monitor

Use the Data Provisioning Agent Monitor to perform basic administration tasks, such as adding adapters to or removing adapters from a Data Provisioning Agent instance.

Prerequisites

The user must have the following roles or privileges to manage adapters:

Table 26: Roles and Privileges

Action	Role or Privilege
Add adapter	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::AddLocationToAdapter • System privilege: ADAPTER ADMIN
Remove adapter	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::RemoveLocationFromAdapter • System privilege: ADAPTER ADMIN
Update adapters	• Role: sap.hana.im.dp.monitor.roles::Operations • System privilege: ADAPTER ADMIN

Context

Use the buttons in the *Agent Monitor* and *Agent Adapter Mapping* tables to perform an action.

Procedure

- To add adapters to an agent instance, select the agent and click *Add Adapters* in the *Agent Monitor* table.
  a. Select the desired adapters from the list of adapters deployed on the agent instance.
  b. Click *Add Adapters*.

  The selected adapters appear in the *Adapter Agent Mapping* table.
- To remove an adapter from an agent instance, select the adapter and click *Remove Location* in the *Adapter Agent Mapping* table.
  a. If the adapter is registered on only one agent instance, you can remove it with *CASCADE option*.
  b. Click *Remove Location*.

  The adapter is removed from the *Adapter Agent Mapping* table.

- To update all adapters for an agent, select the agent and click *Update Adapters* in the *Agent Monitor*. All adapters registered for the selected agent are refreshed, and any new capabilities can be used by SAP HANA.
- To update a single adapter, select the adapter and click *Update* in the *Adapter Agent Mapping* table. The selected adapter is refreshed, and any new capabilities can be used by SAP HANA.

Related Information

CREATE ADAPTER Statement [Smart Data Integration] [page 274]
DROP ADAPTER Statement [Smart Data Integration] [page 290]
Assign Roles and Privileges

5.1.3 Manage Agents Remotely

To manage an agent with the agent configuration tool, you normally must have access to the agent's host operating system. Use the Data Provisioning Agent Management web interface to manage a Data Provisioning agent located within your local network without needing access to the host operating system.

Prerequisites

To use the Data Provisioning Agent management web interface, the following prerequisites must be met:

- You must be on the same local network as the Data Provisioning Agent host.
- The agent management webservice must configured and running on the Data Provisioning Agent host.

Context

The Data Provisioning Agent Management web interface allows you to perform basic management tasks for an agent within your local network:

- Stop and start the agent
- View the agent framework parameters
- View adapter preferences
- Download agent log files

Procedure

1. In a browser, navigate to `https://<agent_hostname>:<port>`

   The default port number is `8050`. If you have configured the management web service with a different port, specify it here.
2. Enter the username and password for your management user.
   The agent status and management options are displayed.
3. Perform any desired management tasks.

Task	Description
Start or stop the agent	Click the *Start* or *Stop* button displayed under the current agent status. Only the button relevant for the current status is displayed.
View agent framework configuration	Expand the *Framework* section of the management interface and click *Show Current Configuration*. The current agent configuration parameters stored in `dpagentconfig.ini` are displayed.
View adapter preferences	Expand the *Adapter Preferences* section of the management interface. 1. Click *Show List of Adapters* to display the adapters on the agent. 2. Select an adapter from the drop-down box. 3. Click *Show Preferences for Selected Adapter*. The currently-configured adapter preferences for the selected adapter are displayed.
Access agent logs	Expand the *Download logs* section of the management interface. 1. Specify the agent directory to access. The *Directory* field is relative to the agent installation directory, and the default is `log` (`<DPAgent_root>/log`). If no value is specified, the contents of the root agent installation directory are displayed. 2. Click *List available logs* to display the files in the specified directory. 3. Select a log from the drop-down box. 4. Click *Download selected log*. The selected log file is downloaded to the local system.

Related Information

Configure the Agent Management Webservice

5.1.4 Back Up the Data Provisioning Agent Configuration

You can back up your Data Provisioning Agent configuration by copying key static configuration files to a secure location. You can use this backup to restore communication between the SAP HANA server and the Data Provisioning Agent.

> **Note**
>
> This backup can be restored only to an agent host with the same fully qualified domain name as the original agent. You can't use the backup to transport configuration settings between agents with different fully qualified domain names.
>
> For example, you can't use a backup from an agent on `<host1>.mydomain.com` to restore settings to an agent on `<host2>.mydomain.com.`

> **Restriction**
>
> Changed-data capture status information for Log Reader adapters can't be backed up and restored.

Unless specified, all files and directories that you need to back up are located under `<DPAgent_root>`:

- `dpagent.ini`
- `dpagentconfig.ini`
- `sec`
- `secure_storage`
- `ssl/cacerts`
- `configuration/com.sap.hana.dp.adapterframework`
- `lib/`
- `camel/`
- `LogReader/config`
- `LogReader/sybfilter/system/<platform>/LogPath.cfg`

5.1.5 Uninstall the Data Provisioning Agent

Uninstall the Data Provisioning Agent from a host system using the uninstallation manager.

For complete information about uninstalling a Data Provisioning Agent from your landscape, see the *Data Provisioning Agent Installation and Update Guide*.

Related Information

Managing the Agent Installation on Linux
Managing the Agent Installation on Windows

5.2 Managing Agent Groups

Agent grouping provides failover and load-balancing capabilities by combining individual Data Provisioning Agents installed on separate host systems.

> ⚠ Restriction
>
> Failover is supported only for real-time requests. For initial or batch load failures that occur because an agent is unavailable, restart the initial or batch load.

> ⚠ Restriction
>
> Load balancing is supported only for initial loads. It is not supported for changed-data capture (CDC) operations.

Planning Considerations

Before configuring agents in a group, review the following considerations and limitations:

- For real-time replication failover, each agent in a group must be installed on a different host system.
- All agents in a group must have identical adapter configurations.
- All agents in a group must use the same communication protocol. You can't mix on-premise agents (TCP) and cloud-based agents (HTTP) in a single group.

Failover Behavior in an Agent Group [page 147]
When an agent node in an agent group is inaccessible for longer than the configured heartbeat interval, the Data Provisioning Server chooses a new active agent within the group. It then resumes replication for any remote subscriptions active on the original agent.

Load Balancing in an Agent Group [page 148]
With multiple agents in an agent group, you can choose to have the agent for the initial loads selected randomly, selected from the list of agents in a round-robin fashion, or not load balanced.

Create or Remove an Agent Group [page 148]
You can create an agent group or remove an existing group using tools available in your SAP HANA landscape.

Manage Agent Nodes in an Agent Group [page 150]
You can manage the agent nodes that belong to an agent group using tools available in your SAP HANA landscape.

Add Adapters to an Agent Group [page 153]
Before you can create remote sources in an agent group, you must add adapters to the group using tools available in your SAP HANA landscape.

Configure Remote Sources in an Agent Group [page 154]
To receive the benefits of failover from an agent group, you must configure your remote sources in the agent group.

Parent topic: Administering Data Provisioning [page 139]

Related Information

Managing Agents and Adapters [page 139]
Managing Remote Sources and Subscriptions [page 157]
Managing Design Time Objects [page 165]
Managing Enterprise Semantic Services [page 174]

5.2.1 Failover Behavior in an Agent Group

When an agent node in an agent group is inaccessible for longer than the configured heartbeat interval, the Data Provisioning Server chooses a new active agent within the group. It then resumes replication for any remote subscriptions active on the original agent.

Initial and batch load requests to a remote source configured on the agent group are routed to the first available agent in the group.

> **⚠ Restriction**
>
> Failover is supported only for real-time requests. For initial or batch load failures that occur because an agent is unavailable, restart the initial or batch load.

Although no user action is required for automatic failover within an agent group, you can monitor the current agent node information.

- Use the following query to return the current master agent name for a remote source:

```
SELECT AGENT_NAME FROM "SYS"."M_REMOTE_SOURCES_" WHERE "REMOTE_SOURCE_OID" =
(SELECT REMOTE_SOURCE_OID FROM "SYS"."REMOTE_SOURCES_" WHERE
REMOTE_SOURCE_NAME = '<remote_source_name>');
```

- Use the following query to generate a list of all agent and agent group names:

```
SELECT AGENT_NAME,AGENT_GROUP_NAME FROM SYS."AGENTS";
```

> **⚠ Caution**
>
> If all nodes in an agent group are down, replication can't continue and must be recovered after one or more agent nodes are available.

Restarting Agent Nodes in an Agent Group

Restarting nodes in an agent group doesn't impact active replication tasks.

For the master agent node, stopping or restarting the agent triggers the failover behavior and a new active master node is selected from the agent group.

5.2.2 Load Balancing in an Agent Group

With multiple agents in an agent group, you can choose to have the agent for the initial loads selected randomly, selected from the list of agents in a round-robin fashion, or not load balanced.

> **Note**
>
> Agent grouping provides load balancing for initial loads only. Load balancing isn't supported for changed-data capture (CDC) operations.

Load balancing is governed by the `'agent_group'.'load_balance_mode'` index server parameter and supports the following modes:

- *none*: No load balancing is performed.
- *random*: The agent is chosen randomly.
- *round_robin*: The chosen agent is the next in the list of available agents after the previously chosen agent.

For example, use the following query to randomly select the agent for initial loads:

```
ALTER SYSTEM ALTER CONFIGURATION ('indexserver.ini','SYSTEM') SET
('agent_group', 'load_balance_mode') = 'random' WITH RECONFIGURE;
```

The default load balancing behavior is *round_robin*.

5.2.3 Create or Remove an Agent Group

You can create an agent group or remove an existing group using tools available in your SAP HANA landscape.

Prerequisites

To create or remove an agent group, you must have the AGENT ADMIN system privilege.

Related Information

CREATE AGENT GROUP Statement [Smart Data Integration] [page 278]
DROP AGENT GROUP Statement [Smart Data Integration] [page 293]

Create or Remove an Agent Group in the Data Provisioning Agent Monitor

Prerequisites

In addition to the AGENT ADMIN system privilege, you must have the following roles or privileges to create or remove an agent group in the Data Provisioning Agent Monitor:

Table 27: Roles and Privileges

Action	Role or Privilege
Create agent group	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::CreateAgentGroup • System privilege: AGENT ADMIN
Remove agent group	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::DropAgentGroup • System privilege: AGENT ADMIN

Context

Use the buttons in the *Agent Group* table to create or remove an agent group.

Procedure

- Click *Create* to create an agent group.

  Specify the name for the new agent group, and click *Create Agent Group*.

  The new agent group appears in the *Agent Group* table.
- Select the agent group and click *Drop* to remove an existing agent group.

  > **Note**
  >
  > When you remove an agent group, any agent nodes for the group are removed from the group first. Agents can't be removed from the group if there are active remote subscriptions.

  Any agent nodes are removed from the group, and the group is removed from the *Agent Group* table.

Create or Remove an Agent Group with SQL Syntax

Context

In a SQL console, use the CREATE or DROP statements to create or remove an agent group.

Procedure

- Use the CREATE statement to create a new agent group:

```
CREATE AGENT GROUP <group_name>
```

- Use the DROP statement to remove an existing agent group:

```
DROP AGENT GROUP <group_name>
```

> **Note**
>
> When you remove an agent group, any agent nodes for the group are removed from the group first. Agents can't be removed from the group if there are active remote subscriptions.

Related Information

CREATE AGENT GROUP Statement [Smart Data Integration] [page 278]
DROP AGENT GROUP Statement [Smart Data Integration] [page 293]

5.2.4 Manage Agent Nodes in an Agent Group

You can manage the agent nodes that belong to an agent group using tools available in your SAP HANA landscape.

Prerequisites

To manage the agent nodes in an agent group, you must have the AGENT ADMIN system privilege.

Related Information

CREATE AGENT Statement [Smart Data Integration] [page 276]
ALTER AGENT Statement [Smart Data Integration] [page 265]

Manage Agent Nodes in the Data Provisioning Agent Monitor

Prerequisites

In addition to the AGENT ADMIN system privilege, you must have the following roles or privileges to manage agent nodes in the Data Provisioning Agent Monitor:

Table 28: Roles and Privileges

Action	Role or Privilege
Create agent	• Role: sap.hana.im.dp.roles::Operations • Application privilege: sap.hana.im.dp.monitor::CreateAgent • System privilege: AGENT ADMIN
Add agent to agent group	• Role: sap.hana.im.dp.roles::Operations • Application privilege: sap.hana.im.dp.monitor::AlterAgent • System privilege: AGENT ADMIN
Remove agent from agent group	• Role: sap.hana.im.dp.roles::Operations • Application privilege: sap.hana.im.dp.monitor::AlterAgent • System privilege: AGENT ADMIN

Context

Use the buttons in the *Agent Monitor* and *Agent Group* tables to perform the action.

> → Tip
>
> Select an agent group in the *Agent Group* table to display its nodes in the *Agent Monitor* table.

Procedure

- To register a new agent with the SAP HANA system and add it to an existing agent group, click *Create Agent*.

  When specifying the parameters for the agent, select the agent group from the *Agent Group* list.

  The new agent appears in the *Agent Monitor* table.

- To modify the group assignment for an existing agent, click *Alter Agent*.
  - Select the new agent group from the *Agent Group* list.
    If you're assigning the agent to a different group, select the empty entry for *Enable SSL* to avoid connection issues when the group is changed.
  - To remove the agent from an agent group, select the empty entry from the *Agent Group* list.

  The group for the agent is displayed in the *Agent Monitor* table.
- To add multiple existing agents to an agent group, select the group in the *Agent Group* table and click *Add Agents*.
  a. Select the agents that you want to add to the group.
  b. Click *Add Agents*.

  The selected agents are assigned to the agent group and all associated entries in the *Agent Monitor* and *Agent Group* tables are updated.

Manage Agent Nodes Using SQL Syntax

Context

In a SQL console, use the CREATE AGENT or ALTER AGENT statements to manage the agent nodes in an agent group.

Procedure

- **Optional:** If you don't know the group name, you can query the system for a list of agent groups:

```
SELECT AGENT_NAME,AGENT_GROUP_NAME FROM SYS."AGENTS";
```

- Use the CREATE AGENT statement to create a new agent in the group:

```
CREATE AGENT <agent_name> PROTOCOL { {'HTTP'} |
                    {'TCP' HOST <agent_hostname> PORT <agent_port_number>}
[{ENABLE | DISABLE} SSL]}  [AGENT GROUP <agent_group_name>]
```

- Use the ALTER AGENT to add an existing agent to the group:

```
ALTER AGENT <agent_name> SET AGENT GROUP <group_name>
```

Related Information

CREATE AGENT Statement [Smart Data Integration] [page 276]
ALTER AGENT Statement [Smart Data Integration] [page 265]

5.2.5 Add Adapters to an Agent Group

Before you can create remote sources in an agent group, you must add adapters to the group using tools available in your SAP HANA landscape.

Prerequisites

To add an adapter to the agent group, you must have the following roles or privileges:

Table 29: Roles and Privileges

Action	Role or Privilege
Add adapter to agent group	System privilege: `ADAPTER ADMIN`

Context

Use the CREATE ADAPTER and ALTER ADAPTER statements to add the adapter to each node in the group.

Procedure

1. **Optional:** If you don't know the agent names, you can query the system for a list of agents and agent groups:

   ```
   SELECT AGENT_NAME,AGENT_GROUP_NAME FROM SYS."AGENTS";
   ```

2. Create the adapter on the first agent node:

   ```
   CREATE ADAPTER "<adapter_name>" AT location agent "<agent1_name>";
   ```

3. Add the adapter to each additional agent node in the agent group:

   ```
   ALTER ADAPTER "<adapter_name>" ADD location agent "<agent#_name>";
   ```

Related Information

CREATE ADAPTER Statement [Smart Data Integration] [page 274]
ALTER ADAPTER Statement [Smart Data Integration] [page 264]

5.2.6 Configure Remote Sources in an Agent Group

To receive the benefits of failover from an agent group, you must configure your remote sources in the agent group.

Related Information

CREATE REMOTE SOURCE Statement [Smart Data Integration] [page 281]
ALTER REMOTE SOURCE Statement [Smart Data Integration] [page 267]

Configure Remote Sources in the SAP HANA Web-Based Development Workbench

Procedure

- To create a new remote source in an agent group:
  a. In the Catalog editor, right-click the *Provisioning* > *Remote Sources* folder, and choose *New Remote Source*.
  b. Enter the required configuration information for the remote source, including the adapter name.
  c. In the *Location* dropdown, choose *agent group*, and select the agent group name.
  d. Click *Save*.
- To add an existing remote source to an agent group:
  a. In the Catalog editor, select the remote source in the *Provisioning* > *Remote Sources* folder.
  b. In the *Location* dropdown, choose *agent group*, and select the agent group name.
  c. Click *Save*.

Related Information

Create a Remote Source in the Web-Based Development Workbench [page 158]

Configure Remote Sources Using SQL Syntax

Context

Use the CREATE REMOTE SOURCE and ALTER REMOTE SOURCE statements to configure remote sources.

Procedure

- Use the CREATE REMOTE SOURCE statement to create a new remote source in the agent group:

```
CREATE REMOTE SOURCE <source_name> ADAPTER <adapter_name> AT LOCATION AGENT
GROUP <group_name> <configuration_clause> <credential_clause>
```

- Use the ALTER REMOTE SOURCE statement to add an existing remote source to the group:

```
ALTER REMOTE SOURCE <source_name> ADAPTER <adapter_name> AT LOCATION AGENT
GROUP <group_name> <configuration_clause> <credential_clause>
```

> → Tip
>
> If you're changing only the location for the remote source, you can omit the ADAPTER and CONFIGURATION clauses:
>
> ```
> ALTER REMOTE SOURCE <source_name> AT LOCATION AGENT GROUP <group_name>
> <credential_clause>
> ```

Related Information

CREATE REMOTE SOURCE Statement [Smart Data Integration] [page 281]
ALTER REMOTE SOURCE Statement [Smart Data Integration] [page 267]

Alter Remote Source Clauses

When you use ALTER REMOTE SOURCE to modify a remote source, you must specify the configuration and credential details as XML strings.

Example Credential Clause

```
WITH CREDENTIAL TYPE 'PASSWORD' USING '<CredentialEntry name="credential">
     <user><username></user>
     <password><password></password>
</CredentialEntry>'
```

Example Configuration Clause

```
CONFIGURATION '<?xml version="1.0" encoding="UTF-8"?>
    <ConnectionProperties name="configurations">
```

```
        <PropertyGroup name="generic">
            <PropertyEntry name="map_char_types_to_unicode">false</PropertyEntry>
        </PropertyGroup>
        <PropertyGroup name="database">
            <PropertyEntry name="cdb_enabled">false</PropertyEntry>
            <PropertyEntry name="pds_use_tnsnames">false</PropertyEntry>
            <PropertyEntry name="pds_host_name"><db_hostname></PropertyEntry>
            <PropertyEntry name="pds_port_number">1521</PropertyEntry>
            <PropertyEntry name="pds_database_name">ORCL</PropertyEntry>
            <PropertyEntry name="cdb_service_name"></PropertyEntry>
            <PropertyEntry name="pds_service_name"></PropertyEntry>
            <PropertyEntry name="pds_tns_filename"></PropertyEntry>
            <PropertyEntry name="pds_tns_connection"></PropertyEntry>
            <PropertyEntry name="cdb_tns_connection"></PropertyEntry>
            <PropertyEntry name="_pds_tns_connection_with_cdb_enabled"></
PropertyEntry>
            <PropertyEntry name="pds_byte_order"></PropertyEntry>
        </PropertyGroup>
        <PropertyGroup name="schema_alias_replacements">
            <PropertyEntry name="schema_alias"></PropertyEntry>
            <PropertyEntry name="schema_alias_replacement"></PropertyEntry>
        </PropertyGroup>
        <PropertyGroup name="security">
            <PropertyEntry name="pds_use_ssl">false</PropertyEntry>
            <PropertyEntry name="pds_ssl_sc_dn"></PropertyEntry>
            <PropertyEntry name="_enable_ssl_client_auth">false</PropertyEntry>
        </PropertyGroup>
        <PropertyGroup name="jdbc_flags">
            <PropertyEntry name="remarksReporting">false</PropertyEntry>
        </PropertyGroup>
        <PropertyGroup name="cdc">
            <PropertyGroup name="databaseconf">
                <PropertyEntry name="pdb_archive_path"></PropertyEntry>
                <PropertyEntry name="pdb_supplemental_logging_level">table</
PropertyEntry>
            </PropertyGroup>
            <PropertyGroup name="parallelscan">
                <PropertyEntry name="lr_parallel_scan">false</PropertyEntry>
                <PropertyEntry name="lr_parallel_scanner_count"></PropertyEntry>
                <PropertyEntry name="lr_parallel_scan_queue_size"></
PropertyEntry>
                <PropertyEntry name="lr_parallel_scan_range"></PropertyEntry>
            </PropertyGroup>
            <PropertyGroup name="logreader">
                <PropertyEntry name="skip_lr_errors">false</PropertyEntry>
                <PropertyEntry name="lr_max_op_queue_size">1000</PropertyEntry>
                <PropertyEntry name="lr_max_scan_queue_size">1000</PropertyEntry>
                <PropertyEntry name="lr_max_session_cache_size">1000</
PropertyEntry>
                <PropertyEntry name="scan_fetch_size">10</PropertyEntry>
                <PropertyEntry name="pdb_dflt_column_repl">true</PropertyEntry>
                <PropertyEntry name="pdb_ignore_unsupported_anydata">false</
PropertyEntry>
                <PropertyEntry name="pds_sql_connection_pool_size">15</
PropertyEntry>
                <PropertyEntry name="pds_retry_count">5</PropertyEntry>
                <PropertyEntry name="pds_retry_timeout">10</PropertyEntry>
            </PropertyGroup>
        </PropertyGroup>
</ConnectionProperties>'
```

> **Note**
>
> You can't change user names while the remote source is suspended.

5.3 Managing Remote Sources and Subscriptions

Remote sources establish the connection between a data provisioning adapter and your source system. Remote subscriptions monitor a remote source for real-time changes to data replicated into the Data Provisioning Server.

Generally, an administrator creates remote sources that can then be used for remote subscriptions in replication tasks and flowgraphs created by a data provisioning modeler.

SAP HANA smart data integration supports parallel operations across remote sources but not within a single remote source. Within a remote source, all operations on the remote source are processed sequentially and you must wait for the completion of one operation before launching another.

Create a Remote Source [page 158]
Using SAP HANA smart data integration, set up an adapter that can connect to your source database, and then create a remote source to establish the connection.

Suspend and Resume Remote Sources [page 162]
You can suspend and resume capture and distribution for remote sources within the *Data Provisioning Remote Subscription Monitor*.

Alter Remote Source Parameters [page 163]
You can modify some remote source parameters while the remote source is suspended.

Manage Remote Subscriptions [page 164]
You can drop, queue, distribute, and reset remote subscriptions within the *Data Provisioning Remote Subscription Monitor*.

Processing Remote Source or Remote Subscription Exceptions [page 165]
Evaluate and resolve remote subscription exceptions.

Parent topic: Administering Data Provisioning [page 139]

Related Information

Managing Agents and Adapters [page 139]
Managing Agent Groups [page 146]
Managing Design Time Objects [page 165]
Managing Enterprise Semantic Services [page 174]

5.3.1 Create a Remote Source

Using SAP HANA smart data integration, set up an adapter that can connect to your source database, and then create a remote source to establish the connection.

Prerequisites

- The user who creates the remote source must have the following roles or privileges:

  Table 30: Roles and Privileges

Action	Role or Privilege
Create a remote source	System privilege: `CREATE REMOTE SOURCE`

- The Data Provisioning Server must be enabled.
- The Data Provisioning Agent must be installed and configured.
- The adapter must be configured and registered with SAP HANA.

Context

You can create a remote source in more than one way.

Related Information

Create a Remote Source in the Web-Based Development Workbench [page 158]
Create a Remote Source Using SQL Syntax [page 159]
Creating Secondary User Credentials [page 161]

5.3.1.1 Create a Remote Source in the Web-Based Development Workbench

In SAP HANA smart data integration, you can create a remote source with the Web-based Development Workbench user interface.

Prerequisites

The user who creates the remote source must have the following roles or privileges:

Table 31: Roles and Privileges

Action	Role or Privilege
Create a remote source	• System privilege: `CREATE REMOTE SOURCE`

Procedure

1. In the Web-based Development Workbench Catalog editor, expand the *Provisioning* node.
2. Right-click the *Remote Sources* folder and choose *New Remote Source*.
3. Enter the required information including the adapter and Data Provisioning Agent names.

   Regarding user credentials, observe the following requirements:
   - A remote source created with a secondary user can be used only for querying virtual tables.
   - If the remote source is used for designing a `.hdbreptask` or `.hdbflowgraph` enabled for real time, use **`technical user`**.
   - If you create a remote subscription using the CREATE REMOTE SUBSCRIPTION SQL statement, use **`technical user`**.
4. Select *Save*.

Related Information

Configure Data Provisioning Adapters
CREATE REMOTE SOURCE Statement [Smart Data Integration] (SAP HANA SQL and System Views Reference) [page 281]

5.3.1.2 Create a Remote Source Using SQL Syntax

In SAP HANA smart data integration, you can create a remote source using SQL syntax.

Prerequisites

To create a remote source, you must have the following roles or privileges:

Table 32: Roles and Privileges

Action	Role or Privilege
Create a remote source	• System privilege: `CREATE REMOTE SOURCE`

Context

To create a remote source using SQL syntax, you must know the connection information for your source. For an existing remote source, the connection information is in an XML string in the CONFIGURATION statement.

Refer to the remote source configuration topic for each adapter in this guide to see sample SQL code. Change the variables to the correct values for your remote source. The example at the end of this topic illustrates a basic CONFIGURATION connection information XML string for a Microsoft SQL Server adapter.

After you create the remote source:

Example: Basic Microsoft SQL Server Remote Source

```
CREATE REMOTE SOURCE "MySQLServerSource" ADAPTER "MssqlLogReaderAdapter" AT
LOCATION AGENT "MyAgent"
CONFIGURATION
'<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConnectionProperties name="configurations">
<PropertyGroup name="data_type_conversion" displayName="Data Type Conversion">
    <PropertyEntry name="map_char_types_to_unicode" displayName="Always Map
Character Types to Unicode">false</PropertyEntry>
    <PropertyEntry name="map_time_to_timestamp" displayName="Map SQL Server Data
Type Time to Timestamp">true</PropertyEntry>
</PropertyGroup>
<PropertyGroup name="database" displayName="Database">
    <PropertyEntry name="pds_server_name" displayName="Host">myserver.sap.corp</
PropertyEntry>
    <PropertyEntry name="pds_port_number" displayName="Port Number">1433</
PropertyEntry>
    <PropertyEntry name="pds_database_name" displayName="Database Name">mydb</
PropertyEntry>
</PropertyGroup>
<PropertyGroup name="cdc" displayName="CDC Properties">
    <PropertyEntry name="pdb_dcmode" displayName="Database Data Capture
Mode">MSCDC</PropertyEntry>
</PropertyGroup>
<PropertyGroup name="logreader" displayName="LogReader">
    <PropertyEntry name="skip_lr_errors" displayName="Ignore log record
processing errors">false</PropertyEntry>
</PropertyGroup>
</ConnectionProperties>
'WITH CREDENTIAL TYPE 'PASSWORD' USING
'<CredentialEntry name="credential">
    <user>myuser</user>
    <password>mypassword</password>
</CredentialEntry>'
```

Next Steps

- If you've recently updated the Data Provisioning Agent, the connection information XML string for your adapter could also have been updated. Refresh the adapter to get the latest connection information.

- To view the connection information for an existing remote source, run the following statement and check the CONNECTION_INFO column:

  ```
  SELECT * FROM "PUBLIC"."REMOTE_SOURCES"
  ```

  > → Tip
  >
  > To ensure you can view the entire XML string in the CONNECTION_INFO column, enable the *Enable zoom of LOB columns* setting in your SAP HANA preferences.

- To view all of the configuration parameters for a given adapter type, run the following statement and check the CONFIGURATION column:

  ```
  SELECT * FROM "PUBLIC"."ADAPTERS"
  ```

  This information can be useful if you want to, for example, determine the name for a given parameter in the user interface, shown as *displayName*.

  ```
  <PropertyEntry name="pds_database_name" displayName="Database
  Name"><database_name></PropertyEntry>
  <PropertyEntry name="pdb_dcmode" displayName="Database Data Capture
  Mode">MSCDC</PropertyEntry>
  ```

Related Information

Configure Data Provisioning Adapters
Update the Data Provisioning Agent
CREATE REMOTE SOURCE Statement [Smart Data Integration] [page 281]

5.3.1.3 Creating Secondary User Credentials

You must use different syntax to create secondary user credentials for SAP HANA smart data integration adapters than the syntax for SAP HANA system adapters.

The syntax for creating secondary user credentials for SAP HANA smart data integration adapters is as follows:

```
create credential for user <user_name> component 'SAPHANAFEDERATION'
purpose <remote_source_name> type 'PASSWORD' using
<CredentialEntry name="credential">
      <user><user_name></user>
      <password><password></password>
</CredentialEntry>
```

5.3.2 Suspend and Resume Remote Sources

You can suspend and resume capture and distribution for remote sources within the *Data Provisioning Remote Subscription Monitor*. For example, suspend your remote sources before maintenance downtime for SAP HANA or your source database, and then resume them after the downtime has ended.

Prerequisites

The user must have the following roles or privileges to suspend and resume capture and distribution:

Table 33: Roles and Privileges

Action	Role or Privilege
Suspend capture or distribution	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::AlterRemoteSource • Object privilege: ALTER on the remote source
Resume capture or distribution	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::AlterRemoteSource • Object privilege: ALTER on the remote source

Context

Use the *Alter Remote Source* button in the monitor to perform the action.

Procedure

1. Select the remote source in the *Remote Source Monitor* table and click *Alter Remote Source*.
2. Click *Suspend* or *Resume* for CAPTURE or DISTRIBUTION.
   Confirmation of the action is displayed in the status console.
3. Close the *Alter Remote Source* dialog.

Results

Capture or distribution for the selected remote source is suspended or resumed.

Related Information

ALTER REMOTE SOURCE Statement [Smart Data Integration] [page 267]
Assign Roles and Privileges
Maintaining Connected Systems [page 199]

5.3.3 Alter Remote Source Parameters

You can modify some remote source parameters while the remote source is suspended.

Context

In the *Installation and Configuration Guide for SAP HANA Smart Data Integration and SAP HANA Smart Data Quality*, see each adapter's remote source description topic regarding which parameters you can modify when a remote source is suspended.

> **Note**
>
> You can't change the User Name parameter when the remote source is suspended.

Procedure

1. In the *Data Provisioning Remote Subscription Monitor*, suspend capture on the remote source.
2. In the SAP HANA Web-based Development Workbench catalog, change the intended remote source parameters.
3. Re-enter the credentials for the remote source and save the changes.
4. Resume capture on the remote source.

Related Information

Configure Data Provisioning Adapters

5.3.4 Manage Remote Subscriptions

You can drop, queue, distribute, and reset remote subscriptions within the *Data Provisioning Remote Subscription Monitor*.

Prerequisites

The user must have the following roles or privileges to manage remote subscriptions:

Table 34: Roles and Privileges

Action	Role or Privilege
Reset, queue, or distribute remote subscription	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::AlterRemoteSubscription • Object privilege: ALTER on the remote subscription
Drop remote subscription	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::DropRemoteSubscription • Object privilege: DROP on the remote subscription

Context

Use the buttons in the *Remote Subscription Monitor* table to perform the action.

Procedure

1. Select the remote subscription in the *Remote Subscription Monitor* table.
2. Click *Queue*, *Distribute*, *Reset*, or *Drop*.

> **Note**
>
> A warning appears if you attempt to drop a remote subscription that is used by any flowgraphs or replication tasks. Click *Drop* if you want to continue and drop the remote subscription anyway.

Results

The remote subscription is queued, distributed, or reset. If you drop a remote subscription, the subscription is removed from the *Remote Subscription Monitor* table.

Related Information

ALTER REMOTE SUBSCRIPTION Statement [Smart Data Integration] [page 271]
CREATE REMOTE SUBSCRIPTION Statement [Smart Data Integration] [page 282]
DROP REMOTE SUBSCRIPTION Statement [Smart Data Integration] [page 294]
Assign Roles and Privileges

5.3.5 Processing Remote Source or Remote Subscription Exceptions

Evaluate and resolve remote subscription exceptions.

If an error occurs or if the row count on the target table doesn't match, for example, look at the Exceptions table and process the entries.

Use the following syntax:

```
PROCESS REMOTE SUBSCRIPTION EXCEPTION <exception_oid> IGNORE|RETRY;
```

To process a remote source or remote subscription exception using the monitoring UI:

1. Click the status of the remote source or remote subscription.
2. Select the error.
3. Click either *Retry Operations* or *Ignore Error.*

Related Information

PROCESS REMOTE SUBSCRIPTION EXCEPTION Statement [Smart Data Integration] [page 297]

5.4 Managing Design Time Objects

Design time objects such as flowgraphs and replication tasks manage the replication and transformation of data in SAP HANA smart data integration and SAP HANA smart data quality.

Execute Flowgraphs and Replication Tasks [page 166]
You can execute design time objects including flowgraphs and replication tasks from the *Data Provisioning Design Time Object Monitor*.

Schedule Flowgraphs and Replication Tasks [page 167]
You can schedule design time objects including flowgraphs and replication tasks within the *Data Provisioning Design Time Object Monitor*.

Stop Non-Realtime Flowgraph Executions [page 169]

You can stop the execution of non-realtime flowgraphs within the *Data Provisioning Design Time Object Monitor*.

Start and Stop Data Provisioning Tasks [page 170]
You can start and stop tasks within the *Data Provisioning Task Monitor*.

Schedule Data Provisioning Tasks [page 171]
You can schedule tasks within the *Data Provisioning Task Monitor*.

Executing Partitions [page 172]
How to execute partitions in a task.

Parent topic: Administering Data Provisioning [page 139]

Related Information

Managing Agents and Adapters [page 139]
Managing Agent Groups [page 146]
Managing Remote Sources and Subscriptions [page 157]
Managing Enterprise Semantic Services [page 174]

5.4.1 Execute Flowgraphs and Replication Tasks

You can execute design time objects including flowgraphs and replication tasks from the *Data Provisioning Design Time Object Monitor*.

> ⚠ Restriction
>
> Real-time flowgraphs and replication tasks can't be executed from the *Data Provisioning Design Time Object Monitor*.

Prerequisites

The user must have the following roles or privileges to schedule flowgraphs and replication tasks.

Table 35: Roles and Privileges

Action	Role or Privilege
Execute flowgraph or replication task	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::ExecuteDesignTimeObject • Object privilege: EXECUTE on the object schema • Object privilege: Any additional object privileges needed within the task (for example, ALTER, CREATE ANY, DELETE, DROP, EXECUTE, INDEX, INSERT, and so on.)

Procedure

1. Select the flowgraph or replication task in the *Design Time Objects* table.
2. Click *Execute*.
   a. If the object uses table type parameters, select the tables to use when executing the object.
   b. If the object uses variable parameters, specify the values to use when executing the object.
   c. Click *Execute*.

Results

The object execution begins and the task appears in the *Task Monitor* table.

Related Information

START TASK Statement [Smart Data Integration] [page 298]
Assign Roles and Privileges

5.4.2 Schedule Flowgraphs and Replication Tasks

You can schedule design time objects including flowgraphs and replication tasks within the *Data Provisioning Design Time Object Monitor*.

> ⚠ Restriction
>
> Real-time flowgraphs and replication tasks can't be scheduled from the *Data Provisioning Design Time Object Monitor*.

Prerequisites

The user must have the following roles or privileges to schedule flowgraphs and replication tasks.

Table 36: Roles and Privileges

Action	Role or Privilege
Enable users to schedule design time objects	Role: sap.hana.xs.admin.roles::JobSchedulerAdministrator
Schedule flowgraph or replication task	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::ScheduleDesignTimeObject

To activate task scheduling, the following must occur:

- Enable scheduling via XS Job Admin Dashboard /sap/hana/xs/admin/jobs/ (The user who enables other users to schedule needs the role sap.hana.xs.admin.roles::JobSchedulerAdministrator).
- To schedule design time objects, the job sap.hana.im.dp.monitor.jobs::scheduleTask needs to be enabled in the *XS Job Details* page: /sap/hana/xs/admin/jobs/#/package/sap.hana.im.dp.monitor.jobs/job/scheduleTask.

Procedure

1. Select the flowgraph or replication task in the *Design Time Objects* table.
2. Click the *Schedules* button.
   The *Schedules* dialog appears.
3. To create a new schedule for the task, click *Add*.
   a. Select the frequency (once or recurring), interval if recurring (year, month, week, day, hour, minute, second), and the time (local, not server time or UTC) for the object execution.
   b. If the object uses table type parameters, select the tables to use when executing the object.
   c. If the object uses variable parameters, specify the values to use when executing the object.
   d. Click *Schedule*.
      The new schedule is added to the list of schedules for the object.
4. To remove an existing schedule, select the schedule and click *Delete.*
   The schedule is removed from the list of schedules for the object.
5. Close the *Schedules* dialog.

Results

Your object executes as scheduled and you can monitor the results of each execution of the object.

Related Information

Assign Roles and Privileges

5.4.3 Stop Non-Realtime Flowgraph Executions

You can stop the execution of non-realtime flowgraphs within the *Data Provisioning Design Time Object Monitor*.

Prerequisites

The user must have the following roles or privileges to stop flowgraph execution.

Table 37: Roles and Privileges

Action	Role or Privilege
Stop flowgraph execution	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::StopTask

Procedure

1. Select the task for the flowgraph in the *Task Monitor* table.
2. Click *Stop*.

Results

The selected flowgraph execution instance is stopped.

Related Information

CANCEL TASK Statement [Smart Data Integration] [page 272]
START TASK Statement [Smart Data Integration] [page 298]

5.4.4 Start and Stop Data Provisioning Tasks

You can start and stop tasks within the *Data Provisioning Task Monitor*.

Prerequisites

The user must have the following privileges to start or stop tasks:

Table 38: Privileges

Action	Privilege
Start task	sap.hana.im.dp.monitor::StartTask
Stop task	sap.hana.im.dp.monitor::StopTask

Procedure

- To start a task, select a task in the *Task Overview* table.
  a. Click *Start*.
  b. If the object uses table type parameters, select the tables to use when executing the object.
  c. If the object uses variable parameters, specify the values to use when executing the object.

  > ⓘ Note
  >
  > Tasks that belong to real-time design time objects can't be started or scheduled from the *Data Provisioning Task Monitor*.

- To stop a task, select the running task in the *Task Execution Monitor* table and click *Stop*.

  Note that there might be a delay in stopping the task depending on when the cancellation was initiated and the pending operation.

Related Information

CANCEL TASK Statement [Smart Data Integration] [page 272]
START TASK Statement [Smart Data Integration] [page 298]
Assign Roles and Privileges

5.4.5 Schedule Data Provisioning Tasks

You can schedule tasks within the *Data Provisioning Task Monitor*.

Prerequisites

The user must have the following roles or privileges to schedule tasks.

Table 39: Roles and Privileges

Action	Role or Privilege
Enable users to schedule task	Role: sap.hana.xs.admin.roles::JobSchedulerAdministrator
Schedule task	• Role: sap.hana.im.dp.monitor.roles::Operations • Application privilege: sap.hana.im.dp.monitor::ScheduleTask

To activate task scheduling, the following must occur:

- Enable scheduling via XS Job Admin Dashboard `/sap/hana/xs/admin/jobs/` (The user that enables other users to schedule needs the role sap.hana.xs.admin.roles::JobSchedulerAdministrator).
- To schedule tasks, the Job sap.hana.im.dp.monitor.jobs::scheduleTask needs to be enabled in the *XS Job Details* page: `/sap/hana/xs/admin/jobs/#/package/sap.hana.im.dp.monitor.jobs/job/scheduleTask`.

Procedure

1. Select the task in the Task Monitor.
2. Click the *Schedules* button.
   The *Schedules* dialog appears.
3. To create a new schedule for the task, click *Add*.
   a. Select the frequency (once or recurring), interval if recurring (year, month, week, day, hour, minute), and the time (local, not server time or UTC) for the task execution.
   b. If the task uses table type parameters, select the tables to use when executing the task.
   c. If the task uses variable parameters, specify the values to use when executing the task.
   d. Click *Schedule*.
      The new schedule is added to the list of schedules for the task.
4. To remove an existing schedule, select the schedule and click *Delete.*
   The schedule is removed from the list of schedules for the task.
5. Close the *Schedules* dialog.

Results

Your task executes as scheduled and you can monitor the results of each execution of the task.

Related Information

Assign Roles and Privileges

5.4.6 Executing Partitions

How to execute partitions in a task.

In the *Data Provisioning Task Monitor*, you can view and execute partitions in the following ways:

- Execute a specific partition.
- Execute partitions that didn't run or failed after executing a task.
- Execute a specific failed partition.

Related Information

Execute a Specific Partition [page 172]
Execute Remaining Partitions [page 173]
Execute a Failed Partition [page 174]
Partition Data in a Replication Task
Partitioning Data in the Flowgraph

5.4.6.1 Execute a Specific Partition

How to execute a specific partition in a task.

Context

In the *Data Provisioning Task Monitor*, you can view and execute specific partitions.

Procedure

1. In the *Task Overview* table, select a task and select *Start*.

   If there are partitions (or table type parameters) configured for the flowgraph or reptask, the *Set Parameters for task* `<name>` displays.
2. Select the partition to execute and select *Start*.

   If no partitions are selected, all partitions execute.
3. Select *Start*.

Example

You can execute a specific partition using the following query:

```
START TASK <task_name> ('_task_execute_partition'=>'<partition_name>');
```

5.4.6.2 Execute Remaining Partitions

How to execute all partitions that didn't run or failed.

Context

After executing a task, in the *Data Provisioning Task Monitor* you can execute all the partitions that didn't run or failed.

Procedure

1. Correct any issues that caused failures.
2. In the *Task Execution Monitor* table, select a failed task execution.
3. Select *Execute Remaining Partitions*.

Example

You can execute remaining partitions using the following query:

```
START TASK <task_name>
('_task_execute_remaining_partitions_since_execution_id'=>'<execution_id>');
```

5.4.6.3 Execute a Failed Partition

How to execute a specific failed partition in a task.

Context

After executing a task, in the *Data Provisioning Task Monitor* you can execute a specific partition that failed.

Procedure

1. Correct any issues that caused failures.
2. In the *Task Execution Monitor* table, for a failed task execution, select the *Partition Count*.

   The *Task Partition Execution Details for Task* `<name>` displays.
3. In the *Task Partition Execution Monitor* table, select the failed partition to execute.
4. Select *Execute Failed Partition*.

Example

You can execute a specific failed partition using the following query:

```
START TASK <task_name> ('_task_execute_partition'=>'<partition_name>');
```

5.5 Managing Enterprise Semantic Services

Use the SAP HANA Enterprise Semantic Services Administration browser-based application to administer and monitor artifacts for semantic services.

To launch the SAP HANA Enterprise Semantic Services Administration tool, enter the following URL in a web browser:

`http://<<your_HANA_instance:port>>/sap/hana/im/ess/ui`

The interface includes the following components (tiles):

Component	Description
Publication Schedules	Publish and unpublish artifacts. Schedule publishing and data profiling requests.

Component	Description
Publication Requests	View the details and status of all requests.
Published Artifacts	View and remove artifacts from the knowledge graph.
Data Profiling Blacklist	Prevent data profiling for selected artifacts.

When you have drilled in to a component, you can click the navigation menu in the upper-left corner to open other components or return to the *Home* page.

Parent topic: Administering Data Provisioning [page 139]

Related Information

Roles for Enterprise Semantic Services [page 175]
Enterprise Semantic Services Knowledge Graph and Publication Requests [page 176]
Publishing Artifacts [page 177]
Monitor the Status of Publication Requests [page 180]
Manage Published Artifacts [page 183]
Data Profiling [page 186]
Setting Configuration Parameters [page 188]
Troubleshooting Enterprise Semantic Services [page 189]
Managing Agents and Adapters [page 139]
Managing Agent Groups [page 146]
Managing Remote Sources and Subscriptions [page 157]
Managing Design Time Objects [page 165]
About SAP HANA Enterprise Semantic Services

5.5.1 Roles for Enterprise Semantic Services

SAP HANA role requirements for Enterprise Semantic Services (ESS).

The following database roles control access to Enterprise Semantic Services.

Description	Role
To use the SAP HANA ESS Administration tool	sap.hana.im.ess.roles::Administrator
To update the SAP HANA ESS configuration	sap.hana.im.ess.roles::Configurator

Description	Role
To use the remote view (sap.hana.im.ess.services.views:REMOTE_OBJECTS) or the lineage table functions (sap.hana.im.ess.services.views.datalineage:GET_ALL_IMPACTING_TABLES, sap.hana.im.ess.services.views.datalineage:GET_IMPACTING_TABLES, sap.hana.im.ess.services.views.datalineage:GET_LINEAGE_FROM_VIEW, sap.hana.im.ess.services.views.datalineage:GET_LINEAGE_FROM_SCHEMA)	sap.hana.im.ess.roles::DataSteward
To use the publishing APIs	sap.hana.im.ess.roles::Publisher
To use the search, ctid API, or the remote view (sap.hana.im.ess.services.views:REMOTE_OBJECTS) or the secured data lineage table functions (sap.hana.im.ess.services.views.datalineage:GET_ACCESSIBLE_LINEAGE_FROM_VIEW, sap.hana.im.ess.services.views.datalineage:GET_ACCESSIBLE_LINEAGE)	sap.hana.im.ess.roles::User

5.5.2 Enterprise Semantic Services Knowledge Graph and Publication Requests

Enterprise Semantic Services enables searching and profiling datasets.

Enterprise Semantic Services uses a knowledge graph that describes the semantics of the datasets that are available to users or applications connected to SAP HANA. It is natively stored in the SAP HANA database.

Datasets represented in the knowledge graph can include tables, SQL views, SAP HANA views, remote objects in remote sources, and virtual tables that refer to remote objects.

An Enterprise Semantic Services publication request extracts information from a resource and publishes it in the knowledge graph. When a user searches for an object based on its metadata and contents, the knowledge graph provides the results.

The knowledge graph becomes populated by one or more of the following methods:

- An SAP HANA administrator uses the Enterprise Semantic Services Administration tool to publish datasets
- An SAP HANA administrator configures the Enterprise Semantic Services REST API so that an application can publish datasets
- If an application has already been configured to call the Enterprise Semantic Services REST API, the application can populate the knowledge graph. For example in SAP HANA Agile Data Preparation, when you add a worksheet, the content publishes to the knowledge graph.

Related Information

Enabling Enterprise Semantic Services
Managing Enterprise Semantic Services [page 174]
SAP HANA Enterprise Semantic Services JavaScript API Reference
SAP HANA Enterprise Semantic Services REST API Reference

5.5.3 Publishing Artifacts

The SAP Enterprise Semantic Services (ESS) Administration tool lets you publish (or unpublish) artifacts.

You can publish or unpublish an artifact programmatically using the on-demand Enterprise Semantic Services API. This method is useful for applications that manage the life cycle of their artifacts, that is, applications that create, delete, and update SAP HANA artifacts. The application determines which artifacts to publish, republish, or unpublish to the Enterprise Semantic Service knowledge graph. An example is the SAP Agile Data Preparation application.

Administrators can also use the SAP HANA ESS Administration tool to publish or unpublish artifacts. This is useful for applications that manage the life cycle of SAP HANA artifacts but do not want (or cannot easily) integrate artifact management with Enterprise Semantic Services. An example is the SAP ERP application. In those cases, it is easier to delegate to an administrator the task of determining which artifacts should be published to Enterprise Semantic Services depending on the needs of application (for example, access to a semantic service like search).

The best practice is to separate the artifacts published by applications using the on-demand ESS API from those published by an administrator using the SAP ESS Administration tool. Therefore, the artifacts will belong to different publisher groups, as shown on the SAP ESS Administration tool Published Artifacts tile.

Related Information

Publish Artifacts [page 178]
Information Available in the Publication Schedule [page 179]
Remove a Schedule for a Deleted Artifact [page 179]
Manage Published Artifacts [page 183]
Information Available in Published Artifacts [page 184]

5.5.3.1 Publish Artifacts

Use the SAP HANA Enterprise Semantic Services (ESS) Administration tool to publish artifacts in the knowledge graph.

Procedure

1. Select the *Publication Schedules* tile.
2. In the Published Artifact Browser, expand the nodes and select an artifact.

   Note that in the following browsers, you can search for an object within a node using a *Filter*: Publication Schedules (catalog only), Published Artifacts, Data Profiling Blacklist, Entity Grid Tags.
   1. Right-click the object and select *Filter*. If the *Filter* option does not display for the object, select *Refresh*.
   2. In the *Filter* dialog box, start typing the object name or part of the name. The list filters as you type.
   3. Select *OK* to save the filter on the node. To clear the filter later, right-click the object and select *Remove filter*.

      > ⓘ Note
      >
      > Selecting *Refresh* on an object removes all filters from the object and its children.
3. To include the artifact, select *Include for publication*.

   Children of the artifact inherit the INCLUDED configuration of the parent unless specifically excluded.
4. Configure the publication schedule as follows.
   a. For *Next Scheduled Date*, click in the box to select a date and time to next publish the artifact.

      If you do not enter a date and time, it is set to the current date and time.
   b. Enter a frequency *Period*.
5. Configure *Data profiling options*.
   a. Select *Discover content type* to include content types in the publication; however, this can impact performance and is not suggested for production scenarios.
   b. Select *Extract searchable values* to extract values; however, this can impact performance and is not suggested for production scenarios.
6. Select *Active* to enable the schedule.

   To deactivate a schedule, clear the checkbox and click *Save*.
7. Select *Save*.

   The schedule displays in the *Schedules* table.
8. To stop (cancel) the publication of an artifact: open Publication Requests monitor, refresh the view, select the *Stop* checkbox for the artifact(s), and select the *Stop* icon in the upper right corner of the window.
9. Optionally, from the navigation menu, open Publication Requests to confirm the status of the request (for example, *REQUEST_PENDING* or *DONE*).
10. To exclude objects, select the object, select *Exclude for publication*, and select *Save*.

Results

The objects appear in the browser tree marked with a solid green plus symbol (for included objects) or a solid red minus symbol (for excluded objects). Inherited objects display an outlined green or red symbol. These markers indicate that a request action has been initiated but is independent of the actual status of the request. To view the status of the request, open the Publication Requests monitor. To view the results of requests, view the Published Artifacts monitor.

5.5.3.2 Information Available in the Publication Schedule

The Publication Schedule displays a table that describes all of the scheduled publication and data profiling requests.

The search field above the table lets you search for published artifacts.

Column Name	Description	Filterable
Publication Artifact	Fully qualified name of the published artifact	Yes
Publication	The mode selected for the schedule (INCLUDE, EXCLUDE, INHERIT)	Yes
Next Scheduled Date	The timestamp for when the schedule will next execute	No
Period	The selected frequency of the schedule	Yes
Active	Whether or not the schedule has been activated	No
Discover	Whether or not the option for *Discover content type* was selected	No
Extracted Searchable Values	Whether or not the option for *Extract searchable values* was selected	No
Warning	A warning indicates that a scheduled artifact has been deleted. The *Delete* option will be enabled on the schedule.	Yes
Delete	To delete a schedule for an artifact that has been deleted, select the checkbox and click *Delete*.	No

5.5.3.3 Remove a Schedule for a Deleted Artifact

Remove a publication schedule for an artifact that was deleted.

Context

When an artifact is configured as included for publication, a publication schedule appears in the table of schedules. If the artifact gets deleted, the publication schedule will remain until the crawler cannot detect it. A warning will then appear for that schedule.

Procedure

1. Select the *Publication Schedules* tile.
2. For the artifact that displays a warning, the *Delete* option will be enabled on that schedule. Select the checkbox and click *Delete* above the column.

5.5.4 Monitor the Status of Publication Requests

Use *Publication Requests* to view and monitor publishing and profiling requests.

Context

To monitor the status of all requests, select the *Publication Requests* tile.

A user can search for only those catalog or remote objects that are described in the knowledge graph as a result of successful publication requests. However, if the name of an artifact unexpectedly does not appear in the search results, the publication of the corresponding artifact might have failed.

Related Information

Information Available on Publication Requests [page 180]

5.5.4.1 Information Available on Publication Requests

Enterprise Semantic Services Publication Requests displays the status and any error messages for each request.

The search field above the table lets you search for published artifacts.

Column Name	Description	Filterable
Detail	Click the magnifying glass icon to see more details. The Detail page displays the following information and statistics: • Request type (see list below) • Submission timestamp • Request status (see list below) • Publisher (user) • Publisher Group • Number of basic artifacts to publish • Number of basic artifacts to unpublish • Number of basic artifacts to profile For the latter three statistics, you can see the associated number of requests that are *Successful*, *Failed*, *In progress*, or *Not started*. The table displays each *Publication Artifact* and its *Publication Status*, *Data Profiling Status*, *Error Code*, and *Error Message* if any. Requests with a FAILED status include a *Retry* checkbox. To retry the request, select the checkbox and click *Retry* at the top of the window. To retry all FAILED requests, select the *Retry* checkbox in the column heading.	No
ID	A number that helps identify a request in the list of requests. This number might not be unique in some cases.	Yes
Request Date	Timestamp of when the request was executed	No
Publication Artifact	Fully qualified name of catalog object or repository object that was published	Yes
Artifact Type	The type of artifact to publish: • SAP HANA views in the repository: attributeview, analyticview, calculationview • SAP HANA catalog objects: table, view, columnview, virtualtable	Yes
Publisher Group	Indicates whether a publication was scheduled using the SAP HANA ESS Administration tool or using the REST API. In the former case, the predefined publisher group is sap.hana.im.ess.AdminPublisherGroup. In the latter case, a call to the publish() API must specify a publisherGroup parameter that defines the ownership of the specified publication in the knowledge graph.	Yes
Publisher	Name of the SAP HANA user who submitted the request	Yes

Column Name	Description	Filterable
Request Type	Request types on the Publication Requests monitor home page include: • ON_DEMAND_PUBLISH • ON_DEMAND_UNPUBLISH • SCHEDULED_PUBLISH • MONITORING_UNPUBLISH • RETRY_ON_DEMAND_PUBLISH • RETRY_ON_DEMAND_UNPUBLISH • RETRY_SCHEDULED_PUBLISH Request types on the artifact Detail page include following: • PUBLISH_NOT_STARTED • UNPUBLISH_NOT_STARTED • UNPUBLISH_IN_PROGRESS • PUBLISH_DONE • PUBLISH_FAILED • UNPUBLISH_FAILED	Yes
Status	Status values on the Publication Requests monitor home page include: • REQUEST_PENDING • IN_PROGRESS • DONE • DONE_WITH_ERRORS • NOTHING_DONE • STOPPING • STOPPED Status values on the artifact Detail page include: • PROFILING_NOT_STARTED • PROFILING_IN_PROGRESS • PROFILING_DONE • PROFILING_FAILED • INACTIVATED • NOT PROFILABLE • BLACKLISTED • OBSOLETE • PUBLICATION_FAILED • STOPPED	Yes
Error Code	Error code when the status is FAILED. Each range of numbers indicates a specific area as follows: • 100-199: SAP HANA adapter errors • 200-399: Prepare, extract, load, and deploy jobs • 700-799: Miscellaneous	Yes

Column Name	Description	Filterable
Error Message	Description of the error	No
Retry	To retry one or more requests, select the *Retry* checkbox for each, or select the *Retry* checkbox in the column heading to select all failed requests, and select the *Retry* button.	No
Stop	To display requests that are currently in progress, select the Refresh icon to update the *Status* column. To move the latest publication requests to the top of the list, for *Request Date* select *Sort Descending*. To stop (cancel) one or more in-progress requests, select the *Stop* checkbox for each, or select the *Stop* checkbox in the column heading to select all requests, and select the *Stop* button.	No

5.5.5 Manage Published Artifacts

Use Published Artifacts to view the artifacts that have been published to the knowledge graph and remove (unpublish) them.

Context

The knowledge graph describes the semantics of published artifacts (datasets). Metadata crawlers and data profiling requests let you publish artifacts to the knowledge graph. Thereby, applications can search for and locate these objects and their metadata.

There are two ways to publish artifacts to the knowledge graph: The HTTP REST API `publish()` method and the SAP HANA Enterprise Semantic Services Administration tool. If the same artifact gets published by both mechanisms, the artifact is identified in the Published Artifacts monitor as belonging to a corresponding publisher group. Therefore, publisher groups define ownership of specific publications in the knowledge graph.

When an artifact is published with a specific publisher group, it can only be unpublished by that group. If the same artifact has been published with multiple publisher groups, it can only unpublished when all corresponding publisher groups unpublish it. This control helps avoid conflicts between applications and an administrator using the Administration tool. Otherwise, an application could publish an artifact and another application or administrator could unpublish it.

In the case of the HTTP `publish()` API, the publisher group name is specific to the application; for example for SAP HANA Agile Data Preparation, it could be com.sap.hana.im.adp. For the SAP HANA ESS Administration tool, the predefined publisher group name is sap.hana.im.ess.AdminPublisherGroup.

To limit the size of both extracted metadata elements and extracted searchable attribute values in knowledge graph, you can also select artifacts to unpublish.

Procedure

1. Select the *Published Artifacts* tile.
2. Expand the nodes on the Published Artifact Browser to find the object to view and select it.

   Note that in the following browsers, you can search for an object within a node using a *Filter*: Publication Schedules (catalog only), Published Artifacts, Data Profiling Blacklist, Entity Grid Tags.
   1. Right-click the object and select *Filter*. If the *Filter* option does not display for the object, select *Refresh*.
   2. In the *Filter* dialog box, start typing the object name or part of the name. The list filters as you type.
   3. Select *OK* to save the filter on the node. To clear the filter later, right-click the object and select *Remove filter*.

      > ⓘ Note
      >
      > Selecting *Refresh* on an object removes all filters from the object and its children.
3. Select the *Publisher Group* as necessary.
4. The table displays all of the published artifacts, when they were last refreshed, and the number of metadata elements in each.
5. To remove an artifact and its data profiling information, select its *Unpublish* checkbox and click *Save*. To unpublish all displayed artifacts, select the *Unpublish* checkbox in the column heading and click *Save*.

   View the *Publication Requests* monitor to confirm that the object was removed. For example, the Request Type would indicate MONITORING_UNPUBLISH.

Related Information

Information Available in Published Artifacts [page 184]

Information Available on Publication Requests [page 180]

5.5.5.1 Information Available in Published Artifacts

Enterprise Semantic Services Published Artifacts displays artifacts that have been published and also lets you remove (unpublish) artifacts from the knowledge graph.

The Published Artifact Browser displays all the published objects available in the Catalog, Content, and Remote Sources folders. The size of an artifact is measured as the total number of searchable metadata elements and searchable attribute values extracted from that artifact.

The search field above the table lets you search for published artifacts.

Name	Description	Filterable
Publisher Group	Indicates whether a publication was scheduled using the SAP HANA ESS Administration tool or using the REST HTTP `publish()` API. When an artifact is published with a specific publisher group, the artifact can only be unpublished by the same group. If a same artifact has been published with different publisher groups, the artifact will be unpublished when all associated groups have unpublished it. For the SAP HANA ESS Administration tool, the predefined publisher group name is sap.hana.im.ess.AdminPublisherGroup. For the REST HTTP `publish()` API, a call to the `publish()` API must specify a `publisherGroup` parameter that determines the name of the publisher group.	Not applicable
Publication Artifact name	Name of the selected artifact.	Not applicable
Number of published artifacts	Number of basic artifacts recursively contained in the selected artifact when the selected artifact is a container. If the selected artifact is a basic artifact, the number of published artifacts is equal to 1.	Not applicable
Number of metadata elements	Total number of extracted metadata elements in the selected artifact.	Not applicable
Number of extracted values	Total number of attribute values extracted in the selected artifact.	Not applicable
Publication Artifact	Qualified name of an artifact that was published or contains published artifacts. The fully qualified name is described by three attributes: • Origin: Catalog or Content • Container: Schema name, package path, or virtual container name • Artifact: Name of the artifact (basic or container) in its parent container	Yes Wildcards
Oldest Refresh	Oldest date of updated basic artifacts in the corresponding container. This date is NULL in the case of a basic artifact.	No
Last Refresh	Most recent date of updated basic artifacts in the corresponding container. This date is the last update in the case of a basic artifact.	No
Basic Artifacts	Number of published basic artifacts recursively contained in the corresponding container. This value is 1 in the case of a basic artifact.	Yes

Name	Description	Filterable
Removable Metadata	Number of non-shared metadata elements. It indicates the number of searchable metadata elements extracted from the corresponding published artifacts that are not shared with other published artifacts. This number gives an indication of how many metadata elements would be removed if you unpublished the artifact.	Yes
Removable Values	Number of searchable attribute values extracted for the catalog object represented by the published artifact. It indicates the number of metadata elements that are not shared with other published artifacts. This number gives an indication of how many profiled values would be removed in the case of unpublishing.	Yes
Unpublish	To unpublish the artifact, select the *Unpublish* check box and click *Save*. To unpublish all displayed artifacts, select the *Unpublish* check box in the column heading and click *Save*.	No

5.5.6 Data Profiling

Enterprise Semantic Services can profile the contents of artifacts that have been published to the knowledge graph.

Data profiling is a process that analyzes the values contained in specific columns of a dataset (the columns to analyze are specified internally using ESS logic). Analysis of the data in a column discovers business types, and searchable values can then be extracted and indexed using SAP HANA full text index.

Enterprise Semantic Services can profile the contents of the following artifacts:

- SQL tables
- Column views issued from graphical Calculation views, Attribute, and Analytic views
- Virtual tables created from remote objects of a remote source with the PASSWORD credential type (see the topic “CREATE REMOTE SOURCE” in the *SAP HANA SQL and System Views Reference*).

> **Note**
>
> When requesting profiling of a catalog object that does not result from an activation, you must assign the role SELECT with grant option to the technical user _HANA_IM_ESS. (For activated objects, there is nothing to do.)

Current limitations include the following:

- Calculation views with dynamic privileges will not be profiled
- Views with required parameters will not be profiled
- Calculation views with dependencies to other views will only be profiled if they are referenced by exactly the same set of analytic privileges as their dependent views. It is advised in this version to only create a single analytic privilege that references all views. Future versions will handle dependencies with different privileges.

Related Information

Limiting Objects to Profile [page 187]
Information Available on the Data Profiling Blacklist [page 188]

5.5.6.1 Limiting Objects to Profile

You can prevent artifacts (limited to catalog objects) from being profiled.

Context

Limiting the artifacts to profile lets you control the volume of searchable attribute values or avoid extracting searchable values from datasets that hold sensitive or personal data.

To prevent an artifact from being profiled, an administrator can blacklist artifacts. When a catalog object that was previously profiled is blacklisted, all of its extracted searchable attribute values are immediately removed from the knowledge graph. The catalog object will never be profiled again, even if there is still a data profiling schedule associated with the object(s).

To blacklist an artifact, follow these steps:

Procedure

1. Select the *Data Profiling Blacklist* tile.
2. In the *Catalog Object Browser*, expand the nodes and select an artifact to blacklist. You can also select a schema to list all its children then select objects to blacklist from within it.

   Note that in the following browsers, you can search for an object within a node using a *Filter*: Publication Schedules (catalog only), Published Artifacts, Data Profiling Blacklist, Entity Grid Tags.
   1. Right-click the object and select *Filter*. If the *Filter* option does not display for the object, select *Refresh*.
   2. In the *Filter* dialog box, start typing the object name or part of the name. The list filters as you type.
   3. Select *OK* to save the filter on the node. To clear the filter later, right-click the object and select *Remove filter*.

      > ⓘ Note
      >
      > Selecting *Refresh* on an object removes all filters from the object and its children.
3. To blacklist the artifact, select the *Blacklisted* check box and click *Save*. To blacklist all displayed artifacts, select the *Blacklisted* check box in the column heading and click *Save*.

   To re-enable data profiling for an artifact, clear the check box and click *Save*.

5.5.6.2 Information Available on the Data Profiling Blacklist

The Enterprise Semantic Services Data Profiling Blacklist lets you view and choose which artifacts to blacklist (remove data profiling values).

The search field above the table lets you search for published artifacts.

Name	Description	Filterable
Catalog Object	Fully qualified name of the selected object	Not applicable
Blacklisted catalog objects	Number of blacklisted objects in the selected artifact	Not applicable
Extracted values	The total number of extracted values for the selected object	Not applicable
Schema Name	Name of the schema to which the artifact belongs	Yes Wildcards
Catalog Object	Catalog object name in the schema	Yes
Extracted Values	Number of extracted searchable values for the object	Yes
Blacklisting Date	Timestamp of when the object was blacklisted	No
Blacklisted	Select the checkbox to blacklist the object and click *Save*. Clear the checkbox to enable data profiling for the object and click *Save*.	No

5.5.7 Setting Configuration Parameters

As an Enterprise Semantic Services administrator, you can set configuration parameter values such as maximum sizes of persistent queues, rolling policy of persistent queues, and so on.

To set configuration parameters, in the SAP HANA studio Administration Console, a system administrator sets values in the reserved table sap.hana.im.ess.eg.configuration::CONFIGURATION. To do so, specific database procedures and user privileges are required.

Example

To set the parameters for "'MAX_ESS_PROFILING_JOBS_SCHEDULE_TIME":

1. Re-enable _HANA_IM_ESS technical user: `ALTER USER _HANA_IM_ESS ENABLE PASSWORD LIFETIME;`
2. Connect with the _HANA_IM_ESS user with its password (the one used during installation).
3. Execute the procedure to increase configuration parameter MAX_ESS_PROFILING_JOBS_SCHEDULE_TIME:

   ```
   CALL
   "SAP_HANA_IM_ESS"."sap.hana.im.ess.eg.configuration::SET_CONFIGURATION_VALU
   E"('MAX_ESS_PROFILING_JOBS_SCHEDULE_TIME', value)
   ```

   where VALUE can be up to 1150000 (the default is 500000 ms).
4. When finished, disable the password: `ALTER USER _HANA_IM_ESS DISABLE PASSWORD LIFETIME;`

5.5.8 Troubleshooting Enterprise Semantic Services

Troubleshooting solutions, tips, and API error messages for Enterprise Semantic Services.

Troubleshoot Installation Issues [page 189]
Troubleshoot Enterprise Semantic Services (ESS) installation issues.

Troubleshoot Publishing Issues [page 190]
Troubleshoot Enterprise Semantic Services (ESS) publishing issues.

Troubleshoot Data Profiling Issues [page 190]
Troubleshoot Enterprise Semantic Services (ESS) data profiling issues.

Troubleshoot Search Issues [page 191]
Troubleshoot Enterprise Semantic Services (ESS) search issues.

API Error Messages [page 193]
Troubleshoot Enterprise Semantic Services (ESS) API errors.

Troubleshooting Tips [page 196]
Tips for troubleshooting and preparing diagnostic information for SAP support.

Related Information

5.5.8.1 Troubleshoot Installation Issues

Troubleshoot Enterprise Semantic Services (ESS) installation issues.

Symptom	When importing the HANA_IM_ESS delivery unit (DU), an activation error occurs and appears in the SAP HANA studio job log view.
Solution	Check the job log details in SAP HANA studio. If the error message is not meaningful, then: • Access the diagnosis file for the index server. • From the bottom of the trace log, look for the first `Check results` message, which should indicate the root cause of the activation failure and suggest how to solve it.

Symptom	The ESS DU has been uninstalled using the uninstallation procedure. When you reimport the ESS DU, activation errors occur, showing that dependent objects are not found.
Cause	Activated objects may have dependent objects that do not yet exist and therefore cause an error.
Solution	Check the following: • Verify that all ESS DUs (including the DEMO DU) have been properly uninstalled through the SAP HANA Application Lifecycle Management console. • Verify that all related packages have been deleted (those with a naming of sap.hana.im.ess...); otherwise remove them as follows: • Create a workspace in the *Repositories* tab of SAP HANA studio.

- Remove the packages from there.

5.5.8.2 Troubleshoot Publishing Issues

Troubleshoot Enterprise Semantic Services (ESS) publishing issues.

Symptom	The Publication Requests monitor displays a message that includes the phrase `If the failure repeats, contact SAP support.`
Cause	Transaction serialization failures in concurrency scenarios.
Solution	If the transaction that failed was a publish request, on the Publication Requests monitor for the artifact in question, select the *Retry* check box and the *Retry* button.

Symptom	Publishing requests appear as not processed in the SAP HANA ESS Administration tool's Publication Schedules view. Request Status remains REQUEST PENDING or REQUESTED.
Solution	Rerun the installation script install.html to verify the installation..

Symptom	A publishing request failed (the Request Status is FAILED in the SAP HANA ESS Administration tool Publication Schedules view).
Cause 1	SAP HANA view definition format is not supported.
Solution 1	The user can “upgrade” the format of the view by editing (make a small change such as adding a space) and saving it.
Cause 2	The SAP HANA view is not supported.
Solution 2	No user action.
Cause 3	API error
Solution 3	Invalid arguments have been passed to the API.

Related Information

API Error Messages [page 193]

5.5.8.3 Troubleshoot Data Profiling Issues

Troubleshoot Enterprise Semantic Services (ESS) data profiling issues.

Symptom	Publishing requests have been processed and all ESS background jobs are active, but data profiling requests appear as not processed in the SAP HANA ESS Administration tool Data Profiling Monitor view. Profiling Status remains as REQUEST PENDING.

Cause	Investigate the error as follows: • Enable the trace level for xsa:sap.hana.im.ess to ERROR in the SAP HANA Administration Console • Inspect the latest diagnosis file `xsengine_alert_xxx.trc`. • Check for the following error message: `ESS815=Profiling Internal Error: Script server must be enabled to profile data` Verify whether the script server was created during installation. To do so, in the SAP HANA Administration Console, view the *Configuration* tab, and in daemon.ini, expand *scriptserver*.
Solution	See "Configure Smart Data Quality" in the *Installation and Configuration Guide*.

Symptom	A run-time object has the Request Status of FAILED in SAP HANA ESS Administration tool Data Profiling Monitor view. An error with message code `ESS805 "Insufficient privilege - user xxx must have SELECT with GRANT option on xxx"` is returned.
Cause	If the run-time object is not an activated object, then check that the SELECT right on the run-time object has been granted WITH GRANT OPTION to the technical user _HANA_IM_ESS.
Solution	Run the following SQL command in SAP HANA studio: `GRANT SELECT ON <catalog object> TO _HANA_IM_ESS WITH GRANT OPTION`

Symptom	A data profiling request failed (the Request Status is FAILED in the SAP HANA ESS Administration tool Data Profiling Monitor view).
Cause 1	Insufficient privileges to _HANA_IM_ESS.
Solution 1	Rerun the install.html script.
Cause 2	API error
Solution 2	Invalid arguments have been passed to the API. See API Error Messages [page 193].

Related Information

Configure Smart Data Quality

5.5.8.4 Troubleshoot Search Issues

Troubleshoot Enterprise Semantic Services (ESS) search issues.

Symptom	The SAP HANA user 'User' cannot perform a search using the ESS API. An internal server error message is returned to the application.
Cause	Investigate the error as follows: • In the SAP HANA Administration Console, set the trace level for xsa:sap.hana.im.ess to INFO.

See Activate Error Trace for Enterprise Semantic Services [page 197].

- Inspect latest diagnosis file xsengine_alert_xxx.trc.
- Check for the following error message:

```
Error: import: package access failed due to missing authorization (…)
```

This means that the 'User' who is publishing has not been granted the privilege Role: sap.hana.im.ess.roles::User.

Solution	Grant the following privilege using SQL commands: `CALL "_SYS_REPO"."GRANT_ACTIVATED_ROLE"('sap.hana.im.ess.roles::User','user') ;`

Symptom	A search query does not return an expected catalog object that exists in the SAP HANA instance. OR Suggestions do not show an expected term, although that term is associated with a database object in the SAP HANA instance.
Cause	Insufficient privileges.
Solution	Verify the user who is posing the search query has sufficient privileges to access the searchable elements of the expected catalog object as in the following table.

Database object type	Searchable elements	Privileges needed
table, SQL view, virtual table, column view	metadata	Owner, object privilege, READ CATALOG, DATA ADMIN
table, SQL view, virtual table	Profiled data	Owner, SELECT object privilege
column view	Profiled data	Owner, SELECT object privilege, analytic privilege

Symptom	A search query does not return an expected database object that exists in the SAP HANA instance.
Cause	Assuming that the user has sufficient required authorizations, check the syntax of the search query.
Solution	See the following examples of common errors.

Search query	Unmatched searchable element	Correction
Customer name	Does not match “NAME1”	Name* Or follow suggested words
Sales ATT	Does not match value “AT&T”	“AT T” will match AT&T, AT-T, AT/T

Search query	Unexpected match	Correction
Sales_2006	Sales Sales_2007	“sales_2006”
Unit sales	Product unit	“unit sales”

Search query	Unexpected match	Correction
Open insurance contract	Closed insurance contract	+open insurance contract “open insurance contract”

Symptom	Acronyms or abbreviations are not matched by a search query, which, as a result, does not return an expected database object that exists in the SAP HANA instance.
Cause	A configuration is missing in the term-mapping table: SAP_HANA_IM_ESS.”sap.hana.im.ess.services.search::Mapping”
Solution	To modify the entries in the term mapping table, see "Search Term Mapping" in the *Modeling Guide for SAP HANA Smart Data Integration and SAP HANA Smart Data Quality*.

Symptom	A search query does not return an expected database object that exists in the SAP HANA instance.
Cause	Assuming that the user has sufficient authorizations and the syntax of the search query is correct, then use the SAP HANA ESS Administration tool Publish and Unpublish Monitor view to verify whether the database object has been successfully published to ESS, that is, its Request Status is SUCCESSFUL.
Solution	If the Request Status is FAILED, take the appropriate actions according to the error code and message.

5.5.8.5 API Error Messages

Troubleshoot Enterprise Semantic Services (ESS) API errors.

Error	API	Action
ESS100=Error occurred when extracting metadata from a publication artifact.	Publish	Not necessarily an API error. First check trace log file for details on the error.
ESS153=Metadata Extraction Internal Error: No view definition.	Publish	Check API artifact argument or SAP HANA view was concurrently deleted.
ESS154=Metadata Extraction Internal Error: No package name.	Publish	Check API artifact argument or package was concurrently deleted.
ESS158=Package name ‘{0}’ does not exist.	Publish	Check API artifact argument or artifact was concurrently deleted.
ESS159=HANA view ‘{0}/{1}.{2}’ does not exist.	Publish	Check API artifact argument or artifact was concurrently deleted.
ESS160=Publication artifact of type ‘{0}’ is not supported.	Publish	Check API arguments and list of supported types of publication artifacts.

Error	API	Action
ESS161=Schema name '{0}' does not exist.	Publish	Check API artifact argument or schema was concurrently deleted.
ESS162=Catalog object XXX does not exist or is not supported.	Publish	Check API artifact argument or catalog object was concurrently deleted.
ESS163=Invalid publication artifact qualified name '{0}'.	Publish	Not necessarily an API error. Verify the artifact exists.
ESS164=Invalid container path '{0}'. Errors 180 – 186	Publish	Check API artifact argument or updates happened concurrently. Verify the path still exists.
ESS180=Expecting character '{0}' but encountered character '{1}' in the publication artifact.	Publish	Parsing error in artifact name. Check API artifact argument.
ESS181=Close quote without a matching open quote in string '{0}' of the publication artifact.	Publish	Parsing error in artifact name. Check API artifact argument.
ESS182=Invalid catalog publication artifact ['{0}'].	Publish	Parsing error in artifact name. Check API artifact argument.
ESS183=Invalid publication artifact '{0}' ['{1}'].	Publish	Parsing error in artifact name. Check API artifact argument.
ESS184=First identification element of the publication artifact is not a catalog or content '{0}'.	Publish	Parsing error in artifact name. Check API artifact argument.
ESS185=Unmatched quote for identification element {0} in publication artifact.	Publish	Parsing error in artifact name. Check API artifact argument.
ESS186=Identification element {0} should not be empty.	Publish	Parsing error in artifact name. Check API artifact argument.
Error containing string: "Search query syntax error" Errors 500 – 526	Search	Search syntax error. First check explanation given in error message. Then check search query in trace log with trace level DEBUG.

Error	API	Action
Error containing string: “Request error message” Errors 600 – 645	Search	Request message error. First check explanation given in error message. Then check request message in trace log with trace level DEBUG.
ESS705=Invalid scope name [{0}].	Publish	Only roman letters (both lowercase and uppercase), digits, and underscore characters are valid.
ESS715=Invalid type filter [{0}] for container [{1}]. Please use one of [table,virtualtable,view, columnview]	Publish	Invalid argument for typeFilter in API. Check explanation given in error message in trace log.
ESS720=External source [{0}] not supported.	Publish	Check API parameter “source”. Can only be LOCAL.
ESS725=Mandatory argument [{0}] is missing in function [{1}].	Publish CT on-demand Search	Mandatory API argument is missing. First check explanation given in error message. Then check search query in trace log with trace level DEBUG.
ESS730=Invalid value [{0}] for argument [{1}] in function [{2}], expecting: [{3}].	Publish CT on-demand Search	Invalid argument in API. First check explanation given in error message. Then check search query in trace log with trace level DEBUG.
ESS735=Invalid value [{0}] at position [{1}] for argument [{2}] in function [{3}], expecting: [{4}].	Publish CT on-demand Search	Invalid argument in API. First check explanation given in error message. Then check search query in trace log with trace level DEBUG.
ESS808=Profiling Error: Invalid Runtime Object Type {0}. Only the following input values are supported: {1}	CT on-demand	Request to profile an unknown type of run-time object. First check explanation given in error message. Then check search query in trace log with trace level DEBUG.
ESS809=Profiling Error: Column View “{0}”.”{1}” cannot be profiled	CT on-demand	Non-profilable column view due to current restrictions.
ESS810=Profiling Error: {0} “{1}”.”{2}” does not exist. Runtime object does not exist	CT on-demand	Check API argument or artifact was concurrently deleted.
ESS811=Profiling Error: At least one attribute to profile must be given	CT on-demand	Check API argument.

Error	API	Action
ESS812=Profiling Error: “{0}” is not attribute of {1} “{2}”.”{3}”	CT on-demand	Check API argument or artifact has been updated concurrently.

5.5.8.6 Troubleshooting Tips

Tips for troubleshooting and preparing diagnostic information for SAP support.

Related Information

Troubleshoot Repeated Errors [page 196]
Activate Error Trace for Enterprise Semantic Services [page 197]
Open the Error Log [page 198]
Collect Diagnostic Information [page 198]

5.5.8.6.1 Troubleshoot Repeated Errors

Procedure for error messages in HANA ESS Administration Monitor view that include `If the failure repeats`.

Prerequisites

The error trace for Enterprise Semantic Services must be activated before you retry the operation.

Context

To retry the failed operation:

Procedure

1. Go to the Enterprise Semantic Services Administration Monitor by entering the following URL in a web browser:

   `http://<<your_HANA_instance:port>>/sap/hana/im/ess/ui`
2. If the transaction that failed was a publish request, click the Publish and Unpublish Monitor tile, find your run-time object in the *Published Artifact* column, and click the *Retry* icon.

Next Steps

After you retry the operation and it still fails, collect diagnostic information.

5.5.8.6.2 Activate Error Trace for Enterprise Semantic Services

The error trace obtains detailed information about actions in Enterprise Semantic Services.

Prerequisites

To configure traces, you must have the system privilege TRACE ADMIN.

Context

To activate the error trace for Enterprise Semantic Services, follow these steps:

Procedure

1. Log on to SAP HANA studio with a user that has system privilege TRACE ADMIN.
2. In the Administration editor, choose the *Trace Configuration* tab.

3. Choose the *Edit Configuration* button for the trace that you want to configure.
4. Expand the *XS ENGINE* node.

> **Note**
>
> Ensure you select the checkbox *Show All Components*.

5. Locate `xsa:sap.hana.im.ess` and check that *system trace level* is *INFO*, *ERROR*, or *DEBUG*, because it is usually set to *DEFAULT*.
6. Click *Finish*.

5.5.8.6.3 Open the Error Log

After the Enterprise Semantic Services error trace is activated, find the error message.

Procedure

1. Log on to SAP HANA studio with a user name that has system privilege CATALOG READ.
2. In the Administration editor, choose the *Trace Configuration* tab and go to *Diagnosis Files* tab.
3. Look for one of the two most recent files (`xsengine_alert_xxx.trc` or `indexserver_alert_xxx.trc`).
4. Go to the end of the file to see the error.

5.5.8.6.4 Collect Diagnostic Information

Procedure to obtain diagnostic information to send to SAP support.

Procedure

1. Log on to SAP HANA studio with a user name that has the system privilege CATALOG READ.
2. In the Administration editor, choose the *Trace Configuration* tab and go to *Diagnosis Files* tab.
3. Choose *Diagnosis Information* > *Collect*.
4. Send the following information to SAP support:
   - Error message and error code
   - Collected diagnosis files
   - If the error appeared in the Publication Schedules monitor, send all other fields of the displayed row with the error message.

6 Maintaining Connected Systems

Avoid errors and minimize downtime by accounting for SAP HANA smart data integration components when planning maintenance tasks for connected systems such as source databases and the SAP HANA system.

Maintaining the SAP HANA System [page 199]
: Consider any effects to your data provisioning landscape before performing common SAP HANA maintenance tasks.

Maintaining Source Databases [page 204]
: Keep your databases performing at maximum capacity by learning about cleaning log files, recovery, and preparing for restarts.

Related Information

6.1 Maintaining the SAP HANA System

Consider any effects to your data provisioning landscape before performing common SAP HANA maintenance tasks.

Most commonly, suspend and resume any data provisioning remote sources before performing SAP HANA maintenance operations.

Update the SAP HANA System [page 200]
: Suspend and resume remote sources when you must update the SAP HANA system.

Takeover/Failback with SAP HANA System Replication [page 200]
: Suspend and resume remote sources when you must perform a takeover and failback operation with SAP HANA system replication.

Failover with SAP HANA Scale-Out [page 201]
: Suspend and resume remote sources when you must perform a host auto-failover with SAP HANA scale-out.

Migrate Objects with SAP HANA System Copy [page 202]
: Migrate SAP HANA Smart Data Integration objects following an SAP HANA system copy or backup restore.

Parent topic: Maintaining Connected Systems [page 199]

Related Information

Maintaining Source Databases [page 204]

6.1.1 Update the SAP HANA System

Suspend and resume remote sources when you must update the SAP HANA system.

Prerequisites

Be sure to back up the SAP HANA system before starting the upgrade process.

Procedure

1. Suspend all remote sources.
2. Update the SAP HANA system.
3. If needed, update each Data Provisioning Agent in your landscape.
4. Resume all remote sources.

Related Information

SAP HANA Server Installation and Update Guide
Suspend and Resume Remote Sources [page 162]
Update the Data Provisioning Agent

6.1.2 Takeover/Failback with SAP HANA System Replication

Suspend and resume remote sources when you must perform a takeover and failback operation with SAP HANA system replication.

Prerequisites

If the SAP HANA system is used as a data source, both the primary and secondary systems must be configured with a virtual IP.

Procedure

1. Suspend all remote sources.
2. Perform a takeover on the secondary SAP HANA system.
3. Perform a failback on the former primary SAP HANA system.
4. Resume all remote sources.

Next Steps

Following an unplanned failback operation, monitor remote subscription exceptions in the *Data Provisioning Remote Subscription Monitor*. To clear any exceptions, click *Retry Operation*.

Related Information

Suspend and Resume Remote Sources [page 162]

6.1.3 Failover with SAP HANA Scale-Out

Suspend and resume remote sources when you must perform a host auto-failover with SAP HANA scale-out.

Prerequisites

If the SAP HANA system is used as a data source, both the active and standby hosts must be configured with a virtual IP.

Procedure

1. Suspend all remote sources.
2. Stop the active SAP HANA host.
3. Wait for the standby host to take over operations from the primary host.
4. Resume all remote sources.

Next Steps

Following an unplanned failover operation, monitor remote subscription exceptions in the *Data Provisioning Remote Subscription Monitor*. To clear any exceptions, click *Retry Operation*.

Related Information

Setting Up Host Auto-Failover (SAP HANA Administration Guide)
Suspend and Resume Remote Sources [page 162]

6.1.4 Migrate Objects with SAP HANA System Copy

Migrate SAP HANA Smart Data Integration objects following an SAP HANA system copy or backup restore.

Prerequisites

Before performing the SAP HANA system copy, ensure that you have backed up the information needed to recreate SAP HANA Smart Data Integration objects that are not transported automatically.

- Back up connection strings for all your remote sources.
  For more information, see https://launchpad.support.sap.com/#/notes/2459991.
- Export virtual table information and SQL creation statements.
  In SAP HANA Web-based Development Workbench, right-click the object, and choose *Export*.
  Reference KBA 2460723 to verify that you have obtained the correct SQL statement for virtual table creation.

Context

Recreate SAP HANA Smart Data Integration objects on the target SAP HANA system.

Procedure

1. Drop all remote sources.
2. Unset the agent group for all Data Provisioning Agent instances.
3. Drop all Data Provisioning Agents.
4. Drop all agent groups.

5. On the target system, recreate the agent groups, agents, remote sources, and virtual tables using generated SQL scripts.

Related Information

Object Transportability [page 203]

6.1.4.1 Object Transportability

In general, all SAP HANA Smart Data Integration and SAP HANA Smart Data Quality related objects are classified as SAP Core Data Services (CDS) artifacts and SQL artifacts. Transportability depends on the type of each object.

Table 40: CDS Artifacts

Object	Transportable
Virtual table in flowgraph	Yes
Virtual table in replication task	Not applicable. Virtual tables are created during replication task activation; manual creation is not required.
Sequence	Yes
Target table in flowgraph	Yes
Procedure	Yes
Function	Yes
Table type	Yes
Calculation view	Yes
Database view	Yes
Flowgraph	Yes
Replication task	Yes

Table 41: SQL Artifacts

Object	Transportable
Remote source	No
Virtual table in flowgraph	No
Virtual table in replication task	Not applicable. Virtual tables are created during replication task activation; manual creation is not required.

Object	Transportable
Sequence	No For more information, see SAP Note 2414711.
Target table in flowgraph	No

Related Information

Migrate Objects with SAP HANA System Copy [page 202]

6.2 Maintaining Source Databases

Keep your databases performing at maximum capacity by learning about cleaning log files, recovery, and preparing for restarts.

Restart the Source Database [page 205]
: When you must restart a source database, stop any remote source capture and restart the capture after restarting the database.

Change the Source Database User Password [page 205]
: When you change the password for the source database user, you must update any remote sources that access the database with that user.

Cleaning up LogReader Archives [page 206]
: Avoid disk space issues and ensure smooth real-time replication by regularly cleaning up the LogReader archives.

Cleaning up PostgreSQL Source Objects [page 209]
: Run the PostgreSQL cleanup script to clean up any orphaned agent-related objects in the source database.

Recover from Missing LogReader Archives [page 209]
: When archive logs are missing, replication fails. There are multiple solutions for recovering and restarting replication.

Recover from LogReader Database Upgrades [page 210]
: After upgrading a source database, you may need to force a version migration to resume remote sources on LogReader adapters.

Change the Primary Archive Log Path During Replication [page 211]
: Replication isn't impacted when the primary archive log path is changed during replication.

Maintain the Source Database Without Propagating Changes to SAP HANA [page 212]
: Use the Maintenance User Filter to define a source database user that can perform maintenance tasks in a source database, without having the changes propagated to the SAP HANA system through data provisioning adapters.

Recover with Microsoft SQL Server Always On Failover [page 213]

Re-execute a replication task when Microsoft SQL Server fails over during the initial load.

Recover with SAP HANA System Replication Failover [page 213]
Re-execute a replication task when SAP HANA fails over during the initial load.

Parent topic: Maintaining Connected Systems [page 199]

Related Information

Maintaining the SAP HANA System [page 199]

6.2.1 Restart the Source Database

When you must restart a source database, stop any remote source capture and restart the capture after restarting the database.

Procedure

1. Suspend capture on any remote sources that access the database.
2. Restart the source database.
3. Resume capture on the remote sources.

Related Information

Suspend and Resume Remote Sources [page 162]

6.2.2 Change the Source Database User Password

When you change the password for the source database user, you must update any remote sources that access the database with that user.

Procedure

1. Suspend capture on any remote sources that access the database.

2. In the SAP HANA Web-based Development Workbench, locate the remote source and change the password in the remote source credentials properties.
3. Resume capture on any remote sources that access the database.

Related Information

SAP HANA System Replication (SAP HANA Administration Guide)
Suspend and Resume Remote Sources [page 162]
Suspend and Resume Remote Sources [page 162]

6.2.3 Cleaning up LogReader Archives

Avoid disk space issues and ensure smooth real-time replication by regularly cleaning up the LogReader archives.

The Oracle, DB2, and MS SQL LogReader adapters rely on the log reader archive log to retrieve changed data from the source databases. Over time, the archive log can grow to consume large amounts of disk space if it isn't cleaned up.

Clean the Archive Log on Oracle [page 206]
Identify the Log Sequence Number (LSN) and use the RMAN utility to clean the log.

Clean the Archive Log on Microsoft SQL Server [page 207]
Identify the Log Sequence Number (LSN) and truncate the log using a Microsoft SQL Server command.

Clean the Archive Log on DB2 [page 208]
Identify the Log Sequence Number (LSN) and use the DB2 utility to clean the log.

Related Information

6.2.3.1 Clean the Archive Log on Oracle

Identify the Log Sequence Number (LSN) and use the RMAN utility to clean the log.

Procedure

1. Identify the ending truncation point of the last commit transaction in the SAP HANA server.

```
SELECT SRC.REMOTE_SOURCE_NAME, MIN(SUB_M.LAST_PROCESSED_BEGIN_SEQUENCE_ID)
```

```
FROM SYS.M_REMOTE_SUBSCRIPTIONS SUB_M INNER JOIN SYS.REMOTE_SUBSCRIPTIONS_ SUB
ON SUB_M.SUBSCRIPTION_NAME = SUB.SUBSCRIPTION_NAME
INNER JOIN SYS.REMOTE_SOURCES_ SRC
ON SUB.REMOTE_SOURCE_OID = SRC.REMOTE_SOURCE_OID
WHERE LENGTH(LAST_PROCESSED_BEGIN_SEQUENCE_ID) > 0 and SRC.REMOTE_SOURCE_NAME
= '<remote_source_name>' GROUP BY SRC.REMOTE_SOURCE_NAME
```

The ending truncation point is returned. For example:

```
00000000000000000000000000000000000000000000000000000000000000000000000000001298ee5
00000001000003940000670f01e0000001298ee20000000000000000
```

2. Use the ending truncation point to identify the LSN of the log file in Oracle.

```
SQL>SET SERVEROUTPUT ON FORMAT WRAPPED
DECLARE LSN NUMBER(38);
SCN NUMBER(38);
SEQID VARCHAR2(200):='<truncation_point>';
BEGIN
SELECT TO_NUMBER(SUBSTR(SEQID,109,12),'XXXXXXXXXXXXXXX') INTO SCN FROM DUAL;
SELECT SEQUENCE# INTO LSN FROM (SELECT SEQUENCE# FROM V$ARCHIVED_LOG WHERE
ARCHIVED='YES' AND DELETED='NO' AND FIRST_CHANGE#<= SCN AND NEXT_CHANGE# >
SCN) UNION (SELECT SEQUENCE# FROM V$ARCHIVED_LOG WHERE ARCHIVED='YES' AND
DELETED='NO' AND FIRST_CHANGE#<= SCN AND NEXT_CHANGE# > SCN);
DBMS_OUTPUT.PUT_LINE('THE LSN IS: ' || TO_CHAR(LSN));
END;
/
```

The LSN is returned. For example: `916`

3. Identify the number of the Oracle redo log group.

```
SQL> select count(GROUP#) from V$LOG;
```

The number of the Oracle redo log group is returned. For example: `3`

4. Use the RMAN utility to clean the archive log.

The log files that can safely be cleaned have a sequence number smaller than the LSN minus the number of the Oracle redo log group.

For example, with an LSN of `916` and an Oracle redo log group of `3`, logs through sequence number `913` may be safely cleaned.

```
rman target sys/<sys_password>@<SID>
RMAN> delete archive log until sequence <sequence_number>;
```

6.2.3.2 Clean the Archive Log on Microsoft SQL Server

Identify the Log Sequence Number (LSN) and truncate the log using a Microsoft SQL Server command.

Procedure

1. Identify the end LSN of the last commit transaction in the SAP HANA server.

```
SELECT SRC.REMOTE_SOURCE_NAME, MIN(SUB_M.LAST_PROCESSED_BEGIN_SEQUENCE_ID)
```

```
FROM SYS.M_REMOTE_SUBSCRIPTIONS SUB_M INNER JOIN SYS.REMOTE_SUBSCRIPTIONS_ SUB
ON SUB_M.SUBSCRIPTION_NAME = SUB.SUBSCRIPTION_NAME
INNER JOIN SYS.REMOTE_SOURCES_ SRC
ON SUB.REMOTE_SOURCE_OID = SRC.REMOTE_SOURCE_OID
WHERE LENGTH(LAST_PROCESSED_BEGIN_SEQUENCE_ID) > 0 and SRC.REMOTE_SOURCE_NAME
= '<remote_source_name>' GROUP BY SRC.REMOTE_SOURCE_NAME
```

The ending truncation point is returned. For example:

```
00000000000000000000000000000000000000000000000000000000000000000000000032b0000
0ef80004000004010000032b00000ef8000100000ef0000300000000
```

2. Use the truncation point to clean the archive log in MS SQL.

```
declare @seq_id_str varchar(200) = '<>';
declare @xact_seqno_str varchar(200) = substring(@seq_id_str,
len(@seq_id_str) - 34*2+1, 10*2)
declare @xdesid binary(10) = convert(binary(10),'00', 2)
declare @xact_seqno binary(10) = convert(binary(10),@xact_seqno_str, 2)
print 'Commit LSN: '
print @xact_seqno
exec sp_repldone @xdesid, @xact_seqno, 0,0,0
exec sp_replflush
```

6.2.3.3 Clean the Archive Log on DB2

Identify the Log Sequence Number (LSN) and use the DB2 utility to clean the log.

Procedure

1. Identify the current LSN in the DB2 database.

```
db2 => SELECT SUBSTR(RAVALUE,41,12) FROM
 <pds_username>.RA_XLOG_SYSTEM_  WHERE RAKEY = 'ltm_locator'
```

> ⓘ Note
>
> `<pds_username>` is the same user specified in the remote source in SAP HANA.

The LSN is returned. For example: `0000004ff413`

2. Identify the database path in DB2.

```
db2 list active databases
```

The database path is returned.

3. Navigate to the DB2 database path, and identify the archive log file containing the current LSN.

```
db2flsn -q <log_sequence_number>
```

The filename of the log containing the current LSN is returned. For example: `S0000354.LOG`

Archive logs older than this file can safely be cleaned.

4. Identify the first archive log method in DB2.

```
db2 "get db cfg for <database_name>" | grep -E "First log archive method"
```

   a. If the first archive log method is `LOGRETAIN`, use a DB2 command to delete the old log files.

```
db2=> prune logfile prior to <log_file_containing_LSN>
```

   b. If the first archive log method isn't `LOGRETAIN`, delete log files older than the identified LSN log manually from the archive log path on the DB2 host.

6.2.4 Cleaning up PostgreSQL Source Objects

If your PostgreSQL remote subscription is reset on the Data Provisioning Server while the Data Provisioning Agent isn't running, agent-related objects created in the source system may not be cleaned up. Run the PostgreSQL cleanup script to clean up any orphaned agent-related objects in the source database.

On your PostgreSQL source, run the following script:

`LogReader/scripts/postgresql_logreader_cleanup.sql`

6.2.5 Recover from Missing LogReader Archives

When archive logs are missing, replication fails. There are multiple solutions for recovering and restarting replication.

> → Tip
>
> If a secondary archive log path is set before replication, replication with automatically switch to the secondary log path if an archive log is missing from the primary path.
>
> Process any exceptions in the *Data Provisioning Remote Subscription Monitor*.

Procedure

- Restore the missing archive logs from a backup location.

  An exception is raised when an archive log file can't be accessed during replication.

  a. In the *Data Provisioning Remote Subscription Monitor*, click the error in the *Remote Source Monitor* or *Remote Subscription Monitor* tables.
  For example:

```
LogReader is in ERROR state [ORA-00308: cannot open archived log '/
rqa16clnx3_work1/oraclearchive/o12lnxrdb/1_2581_896793021.dbf'
                                ORA-27037: unable to obtain file status
                                Linux-x86_64 Error: 2: No such file or
directory
                                Additional information: 3
```

```
                                    ]. Check LogReader log for details.
Context: null
```

If the error contains the filename of the missing archive log, the log can be restored from a backup. If the error doesn't contain a missing archive log filename, clean the archive log safely.

b. Restore the missing archive log file from the backup location on the source machine.
For example:

```
cp /rqa16clnx3_work1/oraclearchive/backup/1_2581_896793021.dbf /
rqa16clnx3_work1/oraclearchive/o12lnxrdb
```

c. Suspend and resume remote source capture to recover the replication.
d. In the *Data Provisioning Remote Subscription Monitor*, click the error in the CDC status column.
e. Select the error for the adapter, and click *Retry Operation*.
f. Select the error for the receiver, and click *Retry Operation*.
g. In the *Data Provisioning Remote Subscription Monitor*, verify that the CDC status no longer indicates an error.

- Recover when the missing archive logs can't be restored.
  a. Drop all replication tasks with real-time enabled.
  b. Re-create the replication tasks with real-time enabled.

Related Information

Suspend and Resume Remote Sources [page 162]

6.2.6 Recover from LogReader Database Upgrades

After upgrading a source database, you may need to force a version migration to resume remote sources on LogReader adapters.

Context

After a Microsoft SQL Server or Oracle source database is upgraded, the associated log reader adapter may be unable to resume remote sources, and an error such as the following appears in the framework trace log:

```
2018-10-26 15:49:30,403 [ERROR] MssqlRepAgentWrapper |
MssqlRepAgentWrapper.resume  - Failed to resume LogReader. It is in REPLICATE
status.
                                Your request cannot be carried out because the
existing XLog at the primary database <DV1> requires migration.
                                          at
com.sybase.ra.ltm.LTM.startReplication(LTM.java:1079)
```

LogReader adapters store the source database version in an internal table. When a remote source is resumed, the adapter compares the current database version with the stored version and prompts a migration if the versions don't match.

Add a parameter to the Data Provisioning Agent configuration file to force a version migration.

Procedure

1. In the agent configuration file, add the migration parameter for your database type.
   - For Oracle, **`logreader.oracle.migrate=true`**
   - For Microsoft SQL Server, **`logreader.mssql.migrate=true`**

   By default, the agent configuration file is located at `<DPAgent_root>/dpagentconfig.ini`.

   > **Note**
   >
   > No migration is necessary for the DB2 LogReader adapter.

2. Restart the Data Provisioning Agent.
3. For Microsoft SQL Server, if the remote source is configured with "CDC mode"="Native", execute `mssql_server_init.sql` again.
4. Resume the remote source.

Related Information

Suspend and Resume Remote Sources [page 162]

6.2.7 Change the Primary Archive Log Path During Replication

Replication isn't impacted when the primary archive log path is changed during replication.

For example, if a new primary log path is set in Oracle:

```
SQL> SELECT DESTINATION FROM V$ARCHIVE_DEST WHERE DEST_NAME='LOG_ARCHIVE_DEST_1';
DESTINATION
------------------------------------------------------------------------------
D:\oracle12\archive\o12ntpdb1
SQL> ALTER SYSTEM SET LOG_ARCHIVE_DEST_1='LOCATION=D:\oracle12\archive\o12ntpdb1
_2';
System altered.
SQL> SELECT DESTINATION FROM V$ARCHIVE_DEST WHERE DEST_NAME='LOG_ARCHIVE_DEST_1'
;
DESTINATION
------------------------------------------------------------------------------
D:\oracle12\archive\o12ntpdb1_2
```

You can use the *Data Provisioning Remote Subscription Monitor* to verify that the replication for all tables hasn't been impacted.

6.2.8 Maintain the Source Database Without Propagating Changes to SAP HANA

Use the Maintenance User Filter to define a source database user that can perform maintenance tasks in a source database, without having the changes propagated to the SAP HANA system through data provisioning adapters.

The actions that can be filtered depend on the source database:

- For Oracle and Microsoft SQL LogReader adapters, database transactions including INSERT, UPDATE, and DELETE, and DDL changes such as ALTER TABLE
- For the SAP HANA adapter, database transactions such as INSERT, UPDATE, and DELETE

Prerequisites

Determine the source database user that performs the maintenance tasks.

Procedure

1. Suspend any remote sources for the associated source database.
2. In the SAP HANA Web-based Development Workbench, choose *Provisioning* > *Remote Sources* and right-click the remote source.
3. In the *Maintenance User Filter* option, specify the username of the database maintenance user.
4. Re-enter the logon credentials for the source database, and save the changes.
5. Resume the remote sources for the source database.

Related Information

Suspend and Resume Remote Sources [page 162]

6.2.9 Recover with Microsoft SQL Server Always On Failover

Re-execute a replication task when Microsoft SQL Server fails over during the initial load.

Context

If a Microsoft SQL Server failover happens during the initial load of a replication task execution, the replication task fails.

For example:

```
Internal error: Remote execution error occurred when getting next row from
Result Set.
```

> **Note**
>
> If the failover occurs during real-time replication, no action is required, and the replication continues automatically.

Procedure

1. In the *Data Provisioning Design Time Object Monitor* locate the replication task marked as `FAILED`.
2. After the Microsoft SQL Server failover completes, re-execute the replication task.

6.2.10 Recover with SAP HANA System Replication Failover

Re-execute a replication task when SAP HANA fails over during the initial load.

Context

If an SAP HANA failover happens during the initial load of a replication task execution, the replication task fails.

For example:

```
Internal error: sql processing error.
```

> **Note**
>
> If the failover occurs during real-time replication, see Failover with SAP HANA Scale-Out [page 201].

Procedure

1. In the *Data Provisioning Design Time Object Monitor* locate the replication task marked as `FAILED`.
2. After the SAP HANA failover completes, re-execute the replication task.

7 Troubleshooting and Recovery Operations

This section describes common troubleshooting scenarios for your SAP HANA smart data integration landscape, as well as recovery steps to follow when errors occur.

Troubleshooting Real-Time Replication Initial Queue Failures [page 215]
Diagnose and resolve common failure scenarios for initial queues in real-time replication tasks.

Recovering from Replication Failures [page 233]
Replication tasks may fail and generate remote source or subscription exceptions for a number of reasons.

Recovering from Crashes and Unplanned System Outages [page 244]
Resume replication and ensure data consistency when you experience a crash or unplanned system outage.

Troubleshooting Data Provisioning Agent Issues [page 249]
This section describes error situations related to the Data Provisioning Agent and their solutions.

Troubleshooting Other Issues [page 255]
This section describes various issues unrelated to replication failures or the Data Provisioning Agent and their solutions.

Related Information

7.1 Troubleshooting Real-Time Replication Initial Queue Failures

Diagnose and resolve common failure scenarios for initial queues in real-time replication tasks.

Resolve User Privilege Errors [page 216]
Adapters used for real-time replication require the remote source database user to be configured with privileges specific to the source database type.

Resolve Remote Source Parameter Errors [page 217]
A replication task may fail if remote source parameters are specified with invalid or out-of-range values, or if values for any mandatory dependent parameters aren't specified.

Resolve Improper Source Database Configuration [page 218]
Real-time replication tasks may fail when the remote source database hasn't been properly configured to support real-time replication.

Resolve Improper Adapter Configurations on the Agent [page 228]

Real-time replication tasks may fail when the associated adapters haven't been properly configured on the Data Provisioning Agent.

Resolve Uncommitted Source Database Transactions [page 230]
Real-time replication tasks may fail when attempting to queue a remote subscription if the source database or table has uncommitted transactions.

Resolve Log Reader Instance Port Conflicts [page 231]
Log reader adapters require an instance port that must not be used by any other applications on the Data Provisioning Agent host.

Resolve Data Provisioning Server Timeouts [page 232]
When the default message timeout value for the Data Provisioning Server is too short, real-time replication tasks may fail during the initial load operation.

Load Clustered and Pooled Table Metadata into SAP HANA [page 233]
Real-time replication tasks with clustered or pooled tables may fail when metadata hasn't been correctly loaded into the SAP HANA database.

Related Information

7.1.1 Resolve User Privilege Errors

Adapters used for real-time replication require the remote source database user to be configured with privileges specific to the source database type.

Context

Insufficient user privileges are indicated in the `<DPAgent_root>/log/framework.trc` trace log file.

For example:

```
Required privileges and/or roles not granted to the database user [ZMTEST].
Missing privileges and/or roles are [SELECT ANY TRANSACTION]
```

Resolve the error by granting the missing user privileges.

Procedure

1. Connect to the source database with a DBA administrator user.
2. Grant the missing privileges required for the source database to the user specified in the remote source.
3. Re-create or edit the existing replication task.

4. Execute the replication task.

Related Information

Create a DB2 UDB User and Grant Permissions
Create Users and Grant Privileges
SAP HANA Remote Source Configuration
Oracle Database Permissions

7.1.2 Resolve Remote Source Parameter Errors

A replication task may fail if remote source parameters are specified with invalid or out-of-range values, or if values for any mandatory dependent parameters aren't specified.

Context

Remote source parameter errors are indicated in the `<DPAgent_root>/log/framework.trc` log file, and may vary based on the specific remote source parameter.

For example, the following scenarios may cause a remote source parameter error:

- Remote source parameter value is outside the valid range.
  For example, if the Oracle maximum operation queue size is set to a value outside of the valid range, such as 10.

  ```
  [ERROR]
  com.sap.hana.dp.oraclelogreaderadapter.OracleLogReaderAdapter.cdcOpen[669]
  - Adapter validation failed.
  com.sap.hana.dp.cdcadaptercommons.validation.ValidationException: Failed to
  validate properties. Error(s):
  The value [10] of property [lr_max_op_queue_size] is not in the range [25,
  2147483647].
  ```

- An Oracle remote source is configured to use TNSNAMES, but the TNSNAMES file and connection parameters aren't specified.

  ```
  [ERROR] WorkerThread.processRequest  - Adapter validation failed. Failed to
  validate properties. Error(s):
  Property [pds_tns_connection] is mandatory.
  Property [pds_tns_filename] is mandatory. Context: null
  com.sap.hana.dp.adapter.sdk.AdapterException: Adapter validation failed.
  Failed to validate properties. Error(s):
  Property [pds_tns_connection] is mandatory.
  Property [pds_tns_filename] is mandatory.
  ```

- An Oracle remote source is configured as a Multitenant Database, but the container database service name, pluggable database service name, or Oracle multitenant credentials aren't specified.

  ```
  [ERROR] WorkerThread.processRequest  - Adapter validation failed. Failed to
  validate properties. Error(s):
  ```

```
Property [cdb_password] is mandatory.
Property [cdb_service_name] is mandatory.
Property [cdb_username] is mandatory.
Property [pds_service_name] is mandatory. Context: null
com.sap.hana.dp.adapter.sdk.AdapterException: Adapter validation failed.
Failed to validate properties. Error(s):
Property [cdb_password] is mandatory.
Property [cdb_service_name] is mandatory.
Property [cdb_username] is mandatory.
Property [pds_service_name] is mandatory.
```

- A user name is specified incorrectly for a case-sensitive adapter.

```
[ERROR] WorkerThread.processRequest  - Adapter validation failed. User Name
[zmtest] does not exist in the source database. Note that 'User Name' option
is case-sensitive. Context: null
com.sap.hana.dp.adapter.sdk.AdapterException: Adapter validation failed. User
Name [zmtest] does not exist in the source database. Note that 'User Name'
option is case-sensitive.
```

Resolve the error by specifying complete remote source parameter values within the valid range.

Procedure

1. Alter the remote source parameter and specify a value within the valid range or any missing dependent parameter values.
2. Re-create the replication task or edit the existing replication task.
3. Execute the replication task.

Related Information

Configure Data Provisioning Adapters
Alter Remote Source Parameters [page 163]

7.1.3 Resolve Improper Source Database Configuration

Real-time replication tasks may fail when the remote source database hasn't been properly configured to support real-time replication.

Enable the Oracle Archive Log [page 219]
: Real-time replication tasks on Oracle remote sources may fail if the Oracle archive log hasn't been enabled.

Enable Supplemental Logging on Oracle [page 221]
: Real-time replication tasks on Oracle remote sources may fail if supplemental logging on the Oracle source database hasn't been enabled or doesn't match the supplemental logging parameter in the remote source.

Enable the Secure File LOB Setting on Oracle [page 222]
: Oracle LOB data may not replicate correctly when the DB_SECUREFILE setting is set to “ALWAYS” or “PREFERRED”.

Specify the Oracle Service Name [page 222]
: If the Oracle service name isn’t specified correctly, the remote source can be created but remote tables can’t be browsed.

Configure the Microsoft SQL Server Transaction Log [page 223]
: Replication tasks may fail if the Microsoft SQL Server transaction log can’t be read or is full.

Initialize the Microsoft SQL Server Database [page 224]
: Replication tasks may fail when the data capture mode is set to “Native Mode” and the Microsoft SQL Server hasn’t been initialized.

Enable the DB2 Archive Log [page 225]
: Real-time replication tasks on DB2 remote sources may fail if the DB2 archive log hasn’t been enabled.

Create the Temporary Tablespace on DB2 [page 225]
: Real-time replication tasks on DB2 remote sources may fail if the user temporary tablespace hasn’t been created.

Specify a Valid DB2 Port Number [page 226]
: Remote sources for DB2 can’t be created if the DB2 port is outside the allowed range of 1-65335.

Verify the DB2 Native Connection Settings [page 227]
: The LogReader may fail to initialize properly when the DB2 native connection fails.

Define the SAP ASE Adapter Interface [page 227]
: Real-time replication tasks on the SAP ASE adapter may fail if an entry for the adapter hasn’t been added to the interface file of the data server.

Related Information

7.1.3.1 Enable the Oracle Archive Log

Real-time replication tasks on Oracle remote sources may fail if the Oracle archive log hasn’t been enabled.

Context

A missing archive log may be indicated in the `<DPAgent_root>/log/framework.trc` log file.

For example:

```
[ERROR]
com.sap.hana.dp.cdcadaptercommons.StatefulAdapterCDC$CDCOpened.addSubscription[79
1]    - [oracleSrc] Failed to add the first subscription.
SubscriptionSpecification [header=remoteTableId=466, remoteTriggerId=178688,
```

```
sql=SELECT "T1"."INT_C1", "T1"."VARCHAR_C2" FROM """LR_USER"".""TESTTB1"""
"T1" , subscription=, customId=, seqID=SequenceId [value=[0, 0, 0, 0]],
isLastSubscription=true, withSchemaChanges=false, firstSubscriptionOnTable=true,
lastSubscriptionOnTable=true]
com.sap.hana.dp.adapter.sdk.AdapterException: Failed to start Log Reader because
of failure to initialize LogReader. Error:Oracle LogMiner must be installed in
order to use this command. Please verify that LogMiner is installed.
```

An additional error may be indicated in the `<DPAgent_root>/log/repagent.log` log file.

For example:

```
ERROR  com.sybase.ds.oracle.logmnr.LogMiner  Failed to start Log Miner:
STARTSCN=2814017, ENDSCN=2814017
ERROR  com.sybase.ds.oracle.logmnr.LogMiner  Failed to start LogMiner when
verifying LogMiner installation. ORA-01325: archive log mode must be enabled to
build into the logstream
```

Resolve the error by enabling the archive log on the Oracle source database.

Procedure

1. On the Oracle source database, check whether archive logging is enabled.

   ```
   SQL> SELECT LOG_MODE FROM V$DATABASE;
   ```

   If the result for LOG_MODE isn't ARCHIVELOG, archive logging is disabled.
2. Enable archive logging.

   ```
   SQL> SHUTDOWN IMMEDIATE;
   SQL> CONNECT SYS/PASSWORD AS SYSDBA;
   SQL> STARTUP MOUNT;
   SQL> ALTER DATABASE ARCHIVELOG;
   SQL> ALTER DATABASE OPEN;
   ```
3. Specify a local archive log directory.

   ```
   SQL> ALTER SYSTEM SET LOG_ARCHIVE_DEST_1='LOCATION=<local\file\path>';
   ```
4. Verify the archive destination.

   ```
   SQL> SELECT DESTINATION FROM V$ARCHIVE_DEST WHERE
   DEST_NAME='LOG_ARCHIVE_DEST_1';
   DESTINATION
   ```

7.1.3.2 Enable Supplemental Logging on Oracle

Real-time replication tasks on Oracle remote sources may fail if supplemental logging on the Oracle source database hasn't been enabled or doesn't match the supplemental logging parameter in the remote source.

Context

An error may be indicated in the `<DPAgent_root>/log/framework.trc` log file in the following scenarios.

- Minimum database-level supplemental logging is disabled.
  For example:

  ```
  [ERROR]
  com.sap.hana.dp.adapter.framework.core.WorkerThread.processRequest[329]    -
  Adapter validation failed. Minimal supplemental logging not enabled.
  ```

- The remote source parameter is set to "database", but PRIMARY KEY and UNIQUE KEY database-level supplemental logging isn't enabled.
  For example:

  ```
  [ERROR]
  com.sap.hana.dp.adapter.framework.core.WorkerThread.processRequest[329]    -
  Adapter validation failed. Database PRIMARY KEY and/or UNIQUE supplemental
  logging is not enabled.
  ```

- The remote source parameter is set to "table", but table-level supplemental logging isn't enabled.
  For example:

  ```
  [ERROR]
  com.sap.hana.dp.adapter.framework.core.WorkerThread.processRequest[329]    -
  Adapter validation failed. Primary key and/or unique key supplemental
  logging is not turned on for these Oracle system tables: [LOBFRAG$,
  TABPART$, TABSUBPART$, COLTYPE$, INDSUBPART$, MLOG$, TYPE$, INDCOMPART$,
  NTAB$, TABCOMPART$, COLLECTION$, LOB$, PROCEDUREINFO$, SNAP$, OPQTYPE$,
  DEFERRED_STG$, LOBCOMPPART$, ARGUMENT$, RECYCLEBIN$, SEQ$, ATTRIBUTE$,
  INDPART$]
  ```

Resolve the error by enabling a supplemental logging level that matches the remote source configuration.

Procedure

- Enable minimum database-level supplemental logging.

  ```
  ALTER DATABASE ADD SUPPLEMENTAL LOG DATA;
  ```

- Enable PRIMARY KEY and UNIQUE KEY database-level supplemental logging.

  ```
  ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY, UNIQUE) COLUMNS;
  ```

- Enable table-level supplemental logging on each table specified in the error message.

  ```
  ALTER TABLE <TABLE_NAME> ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS;
  ```

For example:

```
ALTER TABLE SYS.ARGUMENT$ ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS;
ALTER TABLE SYS.ARGUMENT$ ADD SUPPLEMENTAL LOG DATA (UNIQUE INDEX) COLUMNS;
ALTER TABLE SYS.ATTRIBUTE$ ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS;
ALTER TABLE SYS.ATTRIBUTE$ ADD SUPPLEMENTAL LOG DATA (UNIQUE INDEX) COLUMNS;
...
```

7.1.3.3 Enable the Secure File LOB Setting on Oracle

Oracle LOB data may not replicate correctly when the DB_SECUREFILE setting is set to “ALWAYS” or “PREFERRED”.

Context

The Oracle Log Reader adapter supports Secure File LOB only when the primary database setting DB_SECUREFILE is set to “PERMITTED”.

Procedure

1. Set the Oracle parameter to “PERMITTED”.

   ```
   SQL> alter system set db_securefile = 'PERMITTED';
   ```

2. Verify the DB_SECUREFILE setting.

   ```
   SQL> show parameter db_securefile
   ```

7.1.3.4 Specify the Oracle Service Name

If the Oracle service name isn’t specified correctly, the remote source can be created but remote tables can’t be browsed.

Context

When the Oracle service name is set as a domain name, the “Database Name” remote source parameter must be specified as the Oracle service name.

For example, when checking the Oracle service name:

```
SQL> show parameter service_names;
NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
service_names                        string      <qualified.domain.name>
```

Procedure

1. In the `listener.ora` file, enable the service listener.

   Set the USE_SID_AS_SERVICE_LISTENER parameter to “ON”.

   For example:

   ```
   USE_SID_AS_SERVICE_LISTENER=ON
   LISTENER =
   (DESCRIPTION_LIST =
       (DESCRIPTION =
       (ADDRESS = (PROTOCOL = TCP)(HOST = <agent_hostname>)(PORT = 1521))
       (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
       )
   )
   ```

2. Restart the Oracle listener.
3. Re-create the replication task.

7.1.3.5 Configure the Microsoft SQL Server Transaction Log

Replication tasks may fail if the Microsoft SQL Server transaction log can’t be read or is full.

Context

- An unreadable transaction log error may be indicated in the `<DPAgent_root>/log/framework.trc` log file.
  For example:

  ```
  ERROR  com.sybase.ds.mssql.log.device.LogDevice  The log file <C:\Program
  Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\DATA\ms2014db3_log.ldf>
  is being locked by SQL Server process.
  ```

  Resolve the error by installing the sybfilter driver and making the log file readable.
- If the Microsoft SQL Server transaction log is full, the following error may be reported:

  ```
  com.microsoft.sqlserver.jdbc.SQLServerException: The transaction log for
  database 'BUDGET_LCL_DTM_HANA' is full due to 'ACTIVE_TRANSACTION'.
  ```

Verify that the auto-extend option for the Microsoft SQL Server log file is enabled, and that the disk where the log file is located has available free space.

Related Information

Make Log Files Readable

7.1.3.6 Initialize the Microsoft SQL Server Database

Replication tasks may fail when the data capture mode is set to “Native Mode” and the Microsoft SQL Server hasn’t been initialized.

Context

The Microsoft SQL Server database must be initialized to ensure that the log reader adapter can open the supplemental log of each table marked for replication.

> **Note**
>
> Each Microsoft SQL Server needs to be initialized only one time.

When the database hasn't been initialized, an error may be indicated.

For example:

```
Error: (256, 'sql processing error: QUEUE: QATUSER_REPTEST_START: Failed to
add subscription for remote subscription QATUSER_REPTEST_START.Error: exception
151050: CDC add subscription failed: Failed to add the first subscription.
Error: Failed to start Log Reader because of failure to initialize LogReader.
Error: The server is not initialized. Please run server_admin init first.\n\n:
line 1 col 1 (at pos 0)')
```

Resolve the error by configuring the primary data server.

Related Information

Configure the Primary Data Server for the First Time

7.1.3.7 Enable the DB2 Archive Log

Real-time replication tasks on DB2 remote sources may fail if the DB2 archive log hasn't been enabled.

Context

A missing archive log may be indicated in the `<DPAgent_root>/log/framework.trc` log file.

For example:

```
[ERROR]
com.sap.hana.dp.adapter.framework.core.WorkerThread.processRequest[329]    -
Failed to add the first subscription. Error: Failed to start Log Reader
because of failure to initialize LogReader. Error: An error occured while
getting DB parameter <LOGARCHMETH1>. Exception: DB2 SQL Error: SQLCODE=-286,
SQLSTATE=42727, SQLERRMC=8192;QARUSER, DRIVER=4.18.60
```

Resolve the error by setting the primary DB2 UDB database transaction logging to archive logging.

Related Information

Verify the Current Archive Setting of the Transaction Log

7.1.3.8 Create the Temporary Tablespace on DB2

Real-time replication tasks on DB2 remote sources may fail if the user temporary tablespace hasn't been created.

Context

A temporary tablespace error may be indicated in the `<DPAgent_root>/log/framework.trc` log file.

For example:

```
[ERROR]
com.sap.hana.dp.adapter.framework.core.WorkerThread.processRequest[329]    -
Failed to add the first subscription. Error: Failed to start Log Reader
because of failure to initialize LogReader. Error: An error occured while
getting DB parameter <LOGARCHMETH1>. Exception: DB2 SQL Error: SQLCODE=-286,
SQLSTATE=42727, SQLERRMC=8192;QARUSER, DRIVER=4.18.60
```

Resolve the error by adding a temporary tablespace to the primary database.

Related Information

Add a Temporary Tablespace to the Primary Database

7.1.3.9 Specify a Valid DB2 Port Number

Remote sources for DB2 can't be created if the DB2 port is outside the allowed range of 1-65335.

Context

An invalid port error may be indicated during remote source creation.

For example:

```
(Catalog) Error reading Remote Object: InternalError:
dberror(CallableStatement.execute): 403 - internal error.: Cannot get remote
source objects: port out of range:70000
```

Resolve the error by configuring DB2 with a port in the valid range (1-65335).

Procedure

1. Change the DB2 port in the services file.
   - On Windows, `C:\Windows\System32\drivers\etc\services`
   - On UNIX or Linux, `/etc/services`
2. Update the database manager configuration.

   ```
   db2> update database manager configuration using svcename [<service_name> |
   <port_number>]
   ```
3. Restart the DB2 server.
4. Re-create the remote source.

7.1.3.10 Verify the DB2 Native Connection Settings

The LogReader may fail to initialize properly when the DB2 native connection fails.

Context

When the native connection fails, you may see the following error in the log:

```
[ERROR]
com.sap.hana.dp.db2logreaderadapter.DB2RepAgentWrapper.initialize[858]    -
Failed to initialize LogReader.
Could not find Resource Bundle containing index: Could not get the log end
locator because: Native database connection failed with code <-1>.
```

Resolve the error by verifying that the connection details for your DB2 database are configured correctly:

- Host
- Port
- Database Name
- Database Source Name

7.1.3.11 Define the SAP ASE Adapter Interface

Real-time replication tasks on the SAP ASE adapter may fail if an entry for the adapter hasn't been added to the interface file of the data server.

Context

An error may be indicated when creating a remote source.

For example:

```
Error reading Remote Object: Internal Error: dberror(CallableStatement.execute):
403 - internal error: Cannot get remote source objects: Unknown streamType:
```

Resolve the error by adding an entry for the SAP ASE adapter to the interface file on the SAP ASE server.

Procedure

1. Add the entry to the interface file on the SAP ASE data server.

   ```
   <entry_name>
   master tcp ether <agent_hostname> <port>
   ```

```
query tcp ether <agent_hostname> <port>
```

> **Note**
>
> The entry name must match the adapter instance name specified in the remote source configuration. The port number must match the SAP ASE adapter server port configured in `<DPAgent_root>/Sybase/interfaces`.

2. Re-execute the replication task.

7.1.4 Resolve Improper Adapter Configurations on the Agent

Real-time replication tasks may fail when the associated adapters haven't been properly configured on the Data Provisioning Agent.

Provide a Compatible JDBC Driver [page 228]
Replication tasks may fail if the JDBC driver isn't provided or doesn't match the version of the source database.

Specify a Compatible Java Runtime Environment [page 229]
Replication errors may occur if a noncompatible Java Runtime Environment (JRE) is provided.

Configure the DB2 Environment Variables [page 229]
Remote subscriptions may fail to queue if the DB2 runtime environment variables aren't set on the Data Provisioning Agent host.

Related Information

7.1.4.1 Provide a Compatible JDBC Driver

Replication tasks may fail if the JDBC driver isn't provided or doesn't match the version of the source database.

Context

When a compatible JDBC driver isn't provided, a replication task failure error may be indicated.

For example:

```
Error reading Remote Object: InternalError: dberror(CallableStatement.execute):
403 - internal error: Cannot get remote source objects: Failed to install and
start Oracle JDBC driver bundle.
```

Resolve the error by providing the correct JDBC driver.

- When available, use the JDBC driver distributed with the source database installation.
  For example:
  - For Oracle, `$ORACLE_HOME/jdbc/lib/ojdbc7.jar`
  - For DB2, `$INSTHOME/sqllib/java/db2jcc4.jar`
- The JDBC driver version shouldn't be lower than the source database version.
  You can use the `java` command to verify the JDBC driver version. For example:

```
java -jar ojdbc7.jar
Oracle 12.1.0.2.0 JDBC 4.1 compiled with JDK7 on Mon_Jun_30_11:30:34_PDT_2014
#Default Connection Properties Resource
#Thu Sep 29 05:37:00 PDT 2016
```

Procedure

1. Copy the JDBC driver to the `<DPAgent_root>/lib` directory.
2. Reopen the replication task and add objects from the remote source.

7.1.4.2 Specify a Compatible Java Runtime Environment

Replication errors may occur if a noncompatible Java Runtime Environment (JRE) is provided.

Context

We recommend that you use the SAP JVM bundled with the Data Provisioning Agent.

For complete information about supported Java Runtime Environment versions, see the *Product Availability Matrix (PAM)*.

7.1.4.3 Configure the DB2 Environment Variables

Remote subscriptions may fail to queue if the DB2 runtime environment variables aren't set on the Data Provisioning Agent host.

Context

A replication failure may be indicated in the `<DPAgent_root>/log/framework.trc` log file.

For example:

```
[INFO ]
com.sap.hana.dp.db2logreaderadapter.DB2RepAgentWrapper.initialize[1186]    -
Initializing LogReader ...
[ERROR]
com.sap.hana.dp.adapter.framework.core.WorkerThread.processRequest[354]    - /
rqac48lnx3_work1/dpfiles/dataprovagent/LogReader/lib/linux64/libsybrauni98.so:
libdb2.so.1: cannot open shared object file: No such file or directory
Context: java.lang.UnsatisfiedLinkError: /rqac48lnx3_work1/dpfiles/dataprovagent/
LogReader/lib/linux64/libsybrauni98.so: libdb2.so.1: cannot open shared object
file: No such file or directory
```

Resolve the error by setting the DB2 runtime environment variables before starting the Data Provisioning Agent.

Procedure

1. Source the DB2 profile in the `<DPAgent_root>/bin/DPAgent_env.sh` file.
   For example, add a line such as the following:

   ```
   source /home/db2inst1/sqllib/db2profile
   ```

   > → Tip
   >
   > By sourcing the profile in the `DPAgent_env.sh` file, the environment variables are set each time you use the Data Provisioning Agent Configuration tool.

2. Restart the agent with the Data Provisioning Agent Configuration tool.
3. Re-execute the replication task.

7.1.5 Resolve Uncommitted Source Database Transactions

Real-time replication tasks may fail when attempting to queue a remote subscription if the source database or table has uncommitted transactions.

Context

A replication task error may be indicated.

For example:

```
SQL ERROR--
Message: ORA-00054: resource busy and acquire with NOWAIT specified or timeout
expired
SQLState: 61000
Remote Code: 54
Message: SQL execution failed
```

Resolve the error by committing any transactions in the source database or table.

Procedure

1. Ensure that any transactions have been committed in the source database or table.
2. Re-execute the replication task.
   For trigger-based adapters such as SAP HANA or Teradata, reset the remote subscription.

7.1.6 Resolve Log Reader Instance Port Conflicts

Log reader adapters require an instance port that must not be used by any other applications on the Data Provisioning Agent host.

Context

Log reader adapter instances can't be created when the specified instance port is already in use.

Procedure

1. Suspend any remote sources.
2. Edit the remote source configuration and specify an instance port that isn't in use on the agent host.
3. Resume any remote sources.
4. In the *Data Provisioning Remote Subscription Monitor*, process any remote subscriptions by choosing *Retry Operations*.

Related Information

Monitoring Remote Subscriptions [page 37]
Suspend and Resume Remote Sources [page 162]

7.1.7 Resolve Data Provisioning Server Timeouts

When the default message timeout value for the Data Provisioning Server is too short, real-time replication tasks may fail during the initial load operation.

Context

The data provisioning adapter used by the replication task reports a timeout error.

For example:

```
Error: (256, 'sql processing error: QUEUE: SUB_T002: Failed to add
subscription for remote subscription SUB_T002[id = 165727] in remote source
MSSQLECCAdapterSrc[id = 165373]. Error: exception 151050: CDC add subscription
failed: Request timed out.\n\n: line 1 col 1 (at pos 0)')
```

The error can be resolved by increasing the message timeout value or cleaning up the source database archive log.

Procedure

- Increase the timeout value by adjusting the `messageTimeout` value.

  ```
  ALTER SYSTEM ALTER CONFIGURATION ('dpserver.ini','SYSTEM') SET ('framework',
  'messageTimeout')= '9999'  WITH RECONFIGURE;
  ```

- Clean the source database archive log.

Related Information

Cleaning up LogReader Archives [page 206]

7.1.8 Load Clustered and Pooled Table Metadata into SAP HANA

Real-time replication tasks with clustered or pooled tables may fail when metadata hasn't been correctly loaded into the SAP HANA database.

Context

Missing metadata is indicated in the `<DPAgent_root>/log/framework.trc` log file.

For example:

```
[ERROR]
com.sap.hana.dp.cdcadaptercommons.StatefulAdapterCDC$CDCOpened.addSubscription[79
1]    - [OracleECCAdapterSrc] Failed to add the first subscription.
SubscriptionSpecification [header=remoteTableId=68, remoteTriggerId=160778,
sql=, subscription=, customId=, seqID=SequenceId [value=[0, 0, 0, 0]],
isLastSubscription=true, withSchemaChanges=false, firstSubscriptionOnTable=true,
lastSubscriptionOnTable=true]
java.lang.NullPointerException: while trying to invoke the method
com.sap.hana.dp.adapter.sdk.parser.Query.getFromClause() of a null object loaded
from local variable 'query'
```

Resolve the error by replicating the metadata into SAP HANA.

> **Note**
>
> Beginning with the ABAP Platform 1808/1809 release, cluster and pooled tables aren't supported in SAP S/4HANA HANA.

Procedure

1. Edit the dictionary replication script with the name of the remote source and the SAP HANA schema name.

   By default, the dictionary replication script is located at `<DPAgent_root>/LogReader/scripts/replicate_dictionary.sql`.
2. Execute the dictionary replication script in the SAP HANA database.
3. Re-execute the failed replication task.

7.2 Recovering from Replication Failures

Replication tasks may fail and generate remote source or subscription exceptions for a number of reasons.

Generally, recovering replication involves processing any exceptions in addition to other tasks.

Check for Log Reader Errors [page 234]

Replication may stop if there are log reader errors.

Recover from a Source Table DDL Schema Change [page 235]
Table replication may fail when there's a DDL schema change in the source table and the replication task isn't configured to replicate with structure.

Recover from a Truncated Source Table [page 236]
A replication task may fail when a source table is truncated.

Recover from Source Table and Replication Task Recreation [page 236]
Restore replication after a failure when a source table and replication task are both re-created in a short timeframe.

Recover from a Source and Target Data Mismatch [page 237]
Replication may fail when a new row is inserted in a source table that has a primary key and the target table already contains the row.

Recover from Data Inconsistencies [page 238]
After a replication task has failed, you can use the lost data tracker in the Data Provisioning Agent command-line configuration tool to identify and correct data inconsistencies that may have occurred.

Recover from an Agent Communication Issue [page 240]
Process any remote subscription exceptions and restore replication when a network interruption or other communication issue occurs between the Data Provisioning Agent and the SAP HANA server.

Resolve Stopped or Delayed Replication on Oracle [page 241]
If replication from an Oracle source system is stopped or delayed, or if you notice poor performance, check the instance log for reports of unsupported transactions.

Resolve Locked SAP HANA Source Tables [page 242]
A replication task or flowgraph may fail if an SAP HANA source table is locked and the remote subscription request times out.

Reset the Remote Subscription [page 242]
When log reader-based real-time replication has stopped, you can try resetting the remote subscription.

Clear Remote Subscription Exceptions [page 244]
Replication may be stopped due to remote subscription exceptions. For example, a remote subscription exception may be generated when a primary key violation, constraint violation, or null primary key occurs.

Related Information

7.2.1 Check for Log Reader Errors

Replication may stop if there are log reader errors.

For more information about log reader errors, check the trace file located in the `log` folder of your Data Provisioning Agent instance. You may also choose to increase the trace log levels for the Data Provisioning Server.

Related Information

Activate Additional Trace Logging for the Data Provisioning Server [page 256]

7.2.2 Recover from a Source Table DDL Schema Change

Table replication may fail when there's a DDL schema change in the source table and the replication task isn't configured to replicate with structure.

Context

When table replication fails, remote source and remote subscription exceptions may be generated.

Recover replication by processing any exceptions and recreating the replication task.

Procedure

1. In the *Data Provisioning Remote Subscription Monitor*, verify any reported exceptions.
   Schema change exceptions on the failed replication task can't be ignored or retried.
2. In the SAP HANA Web-based Development Workbench editor, drop the failed replication task.
3. Re-create the replication task with the "Initial + realtime with structure" replication behavior.
4. Activate and execute the new replication task.
5. In the *Data Provisioning Remote Subscription Monitor*, ignore the schema change exception.

Related Information

Create a Replication Task
Save and Execute a Replication Task

7.2.3 Recover from a Truncated Source Table

A replication task may fail when a source table is truncated.

Context

Restore replication by truncating the target table and processing any remote subscription exceptions.

Procedure

1. In the *Data Provisioning Remote Subscription Monitor*, verify any remote subscription errors.
2. Truncate the target table in the SAP HANA system to restore data consistency between the source and target tables.
3. In the *Data Provisioning Remote Subscription Monitor*, ignore the TRUNCATE TABLE errors.

Related Information

Processing Remote Source or Remote Subscription Exceptions [page 165]

7.2.4 Recover from Source Table and Replication Task Recreation

Restore replication after a failure when a source table and replication task are both re-created in a short timeframe.

Context

A replication failure may occur under the following conditions:

- There are multiple active remote subscriptions to the same source database.
- There's a significant amount of change data on the source table or other subscribed tables that haven't yet been replicated to the SAP HANA target
- The remote subscription or replication task is dropped, and the source table and replication task are re-created in a short timeframe.

Under these circumstances, Log Reader adapters may capture the DROP TABLE DDL and attempt to replicate it to the SAP HANA target, generating an exception.

Restore replication by processing any remote source and subscription exceptions.

Procedure

1. In the *Data Provisioning Remote Subscription Monitor*, check for remote subscription errors.
2. Restore replication by selecting the exception and choosing *Ignore*.

Related Information

Processing Remote Source or Remote Subscription Exceptions [page 165]

7.2.5 Recover from a Source and Target Data Mismatch

Replication may fail when a new row is inserted in a source table that has a primary key and the target table already contains the row.

> **Note**
>
> Updating or deleting source table rows that don't exist in the target table doesn't cause a replication failure.

Context

Recover replication by processing any remote subscription exceptions and re-executing the replication task.

Procedure

1. In the *Data Provisioning Remote Subscription Monitor*, identify any errors.
2. Select the exception and choose *Ignore* to restore replication.
3. (Optional) Re-execute the replication task to avoid any data mismatch between the source and target tables.

Related Information

Processing Remote Source or Remote Subscription Exceptions [page 165]

7.2.6 Recover from Data Inconsistencies

After a replication task has failed, you can use the lost data tracker in the Data Provisioning Agent command-line configuration tool to identify and correct data inconsistencies that may have occurred.

Prerequisites

Before attempting to recover from data inconsistencies, ensure that you have downloaded and installed the correct JDBC libraries. See the *SAP HANA smart data integration Product Availability Matrix (PAM)* for details. Place the files in the `<DPAgent_root>/lib` folder.

> **⚠ Restriction**
>
> The lost data tracker supports only scenarios where the source and target are both SAP HANA.

Context

The tool tracks data inconsistencies by using a specified column as a partition in both the source and target tables, and then comparing data row counts for each partition. Inconsistencies between source and target can be printed as a SQL fix plan, or corrected automatically.

Two fix modes are available:

- `row`: Inconsistencies are processed row by row.
- `block` (Default): Inconsistencies are processed by partition block.

Procedure

1. Configure the source and target database connections.

   Create an INI file with the hostname, port number, database name, username, and password for each database in the following format:

   ```
   source_db_host=<hostname>
   source_db_port=<port_number>
   source_db_name=<database_name>
   source_db_user=<username>
   source_db_pswd=<password>
   target_db_host=<hostname>
   target_db_port=<port_number>
   target_db_name=<database_name>
   target_db_user=<username>
   target_db_pswd=<password>
   ```

> → Tip
>
> Save the configuration file as `<DPAgent_root>\bin\DBConfig.ini` to define the default configuration for the lost data tracker tool.

2. At the command line, navigate to `<DPAgent_root>\bin`.
3. Start the configuration tool with the `--LostDataTracker` parameter.
   - On Windows, `agentcli.bat --LostDataTracker`
   - On Linux, `agentcli.sh --LostDataTracker`
4. Choose *Track Lost Data Configuration*.
5. As prompted, specify configuration parameters for partitioning and tracking depth.

   Press `Enter` to accept the default values or skip optional parameters.
   - Source table name
   - Column name to use for partition blocks.
     For example, a date/time column.
   - Target table name
   - Column name to use as a primary key for row-mode fixes
   - Path to the database configuration file

     > → Tip
     >
     > By default, the tool looks for the configuration file at `<DPAgent_root>/bin/DBConfig.ini`.
   - *Difference threshold* for partition blocks
     If the difference between the source and target partitions is greater than the threshold, inconsistency tracking continues into the partition.
   - *Maximum row scan count*
     When the source and target partition row counts match, and the number of rows is less than the *Maximum scan row count*, the partition is skipped for further tracking.
6. As prompted, specify configuration parameters for inconsistency tracking and data fixing.
   - Tracking mode
     - `row`: Inconsistencies are processed row by row.
     - `block`: Inconsistencies are processed by partition block.

     Default: `block`
   - Whether to generate a SQL fix plan for any inconsistencies
     Default: `yes`
   - Whether to automatically fix any inconsistencies
     Default: `yes`
   - Virtual table name to use when processing fixes

Next Steps

After specifying all configuration parameters, the lost data tracker automatically starts comparing the source and target tables.

If a SQL fix plan was requested, a SQL command for each fix is displayed.

If automatic inconsistency correction was specified, the fix for each inconsistency detected is automatically applied to the target table.

Related Information

SAP HANA smart data integration and all its patches Product Availability Matrix (PAM) for SAP HANA SDI 2.0

7.2.7 Recover from an Agent Communication Issue

Process any remote subscription exceptions and restore replication when a network interruption or other communication issue occurs between the Data Provisioning Agent and the SAP HANA server.

Context

A communication issue may generate remote subscription exceptions such as the following:

- Connection to the agent has been lost.
- Connection to all the agents in the cluster has been lost.
- One or more subscriptions failed for the remote source.

In some scenarios, no exceptions are generated, but changed source data isn't replicated to the target. In these scenarios, an error message may be indicated in the Data Provisioning Server trace log.

For example:

```
Generic stream error: getsockopt, Event=EPOLLERR - , rc=111: Connection refused
$NetworkChannelBase$=
92 [0x00007f054cdb1798]  {refCnt=4, idx=4} 10.173.160.203/0_tcp-
>10.173.160.203/8780_tcp ConnectWait,[r--c]
exception throw location:
 1: 0x00007f23a75ec672 in Stream::NetworkChannelCompletionThread::run(void*&)
+0x420 at
NetworkChannelCompletion.cpp:548 (libhdbbasis.so)
 2: 0x00007f23a752a047 in Execution::Thread::staticMainImp(void**)+0x743 at
Thread.cpp:463 (libhdbbasis.so)
 3: 0x00007f23a752b698 in Execution::Thread::staticMain(void*)+0x34 at
ThreadMain.cpp:26 (libhdbbasis.so)
[58457]{-1}[-1/-1] 2016-11-03 00:33:53.499182 e Stream
NetworkChannelCompletion.cpp(00622) : NetworkChannelCompletionThread #5
91 [0x00007f054cdb1958]  {refCnt=4, idx=5} 10.173.160.203/0_tcp-
>10.173.160.203/8780_tcp ConnectWait,[r--c]
: Error in asynchronous stream event: exception  1: no.2110001  (Basis/IO/Stream/
impl/NetworkChannelCompletion.cpp:548)
```

Procedure

1. Suspend all remote sources on the affected agent.
2. Restart the Data Provisioning Agent.
3. Check whether the status of the agent TCP port is “LISTEN” or “ESTABLISHED”.
   - On Windows, `netstat -na | findstr "5050"`
   - On Linux, `netstat -na|grep 5050`

   Additionally, verify that no operating system firewall rules are configured for the agent TCP port.
4. Restart the Data Provisioning Server on the SAP HANA system.
5. Resume all sources on the affected agent.
6. In the *Data Provisioning Remote Subscription Monitor*, retry or ignore any remaining remote subscription exceptions.
7. Verify the remote source and remote subscription statuses.
   - Remote source CDC status: OK
   - Remote subscription state: APPLY_CHANGE_DATA
8. Check for data consistency by comparing the source and target table row counts.

Related Information

Manage the Agent Service
Operations on Services (SAP HANA Administration Guide)

7.2.8 Resolve Stopped or Delayed Replication on Oracle

If replication from an Oracle source system is stopped or delayed, or if you notice poor performance, check the instance log for reports of unsupported transactions.

Context

When the Oracle LogMiner starts scanning from the middle of a transaction and fails to translate the raw record, it reports an unsupported operation. This unsupported operation occurs most often on UPDATE operations involving wide tables.

The Oracle Log Reader adapter can manage these records by using a standby scanner, but frequent occurrence of unsupported operations can slow scan performance.

You have several options to reduce the number of unsupported operations.

- Upgrade to a newer version of the Data Provisioning Agent, if possible.

- If parallel scanning is enabled, increase the value of the *Parallel scan SCN range* parameter.
- Disable parallel scanning.

Related Information

Data Provisioning Agent Log Files and Scripts [page 249]

7.2.9 Resolve Locked SAP HANA Source Tables

A replication task or flowgraph may fail if an SAP HANA source table is locked and the remote subscription request times out.

Context

A replication task or flowgraph that has timed out due to a locked table may fail with the following error:

```
SAP DBTech JDBC: [256]: sql processing error: QUEUE: RT_SFC100: Failed to add
subscription for remote subscription RT_SFC100.Error: exception 151050: CDC add
subscription failed: Request timed out.
```

Additionally, you can verify a locked table by checking the blocked transactions in the SAP HANA source.

To resolve a locked SAP HANA source table, refer to SAP Note 0001999998.

7.2.10 Reset the Remote Subscription

When log reader-based real-time replication has stopped, you can try resetting the remote subscription.

Procedure

1. From a SQL console, reset the remote subscription.

   ```
   ALTER REMOTE SUBSCRIPTION <subscription_name> RESET;
   ```

   → Tip

   If you have the required privileges, you can also reset remote subscriptions from the *Data Provisioning Remote Subscription Monitor*. For more information, see *Manage Remote Subscriptions*.

2. Check the Data Provisioning Agent for the adapter instance.

   In `<DPAgent_root>/LogReader`, look for a directory with the same name as your remote subscription instance. Back up and delete the folder, if it exists.

3. Execute the cleanup script for your database type.

   The cleanup scripts are located in `<DPAgent_root>/LogReader/scripts.`

   > ⓘ Note
   >
   > Execute the script using the same user that is configured for replication.

4. Stop the SAP HANA Data Provisioning Agent service.

   - On Windows, use the Services manager in Control Panel.
   - On Linux, run `./dpagent_servicedaemon.sh stop`.

5. (Optional) Enable additional logging on the Data Provisioning Agent.

   a. Open the Agent configuration file in a text editor.

      The configuration file is located at `<DPAgent_root>/dpagentconfig.ini`.

   b. Change the parameter `framework.log.level` from INFO to ALL.

   Increasing the log level generates additional information useful for further debugging, if necessary. You can safely revert the log level after resolving the issue.

6. Restart the SAP HANA Data Provisioning Agent service.

   - On Windows, use the Services manager in Control Panel.
   - On Linux, run `./dpagent_servicedaemon.sh start`.

7. From a SQL console, queue and distribute the remote subscription.

```
ALTER REMOTE SUBSCRIPTION <subscription_name> QUEUE
```

```
ALTER REMOTE SUBSCRIPTION <subscription_name> DISTRIBUTE
```

   > → Tip
   >
   > If you have the required privileges, you can also queue and distribute remote subscriptions from the *Data Provisioning Remote Subscription Monitor*. For more information, see *Manage Remote Subscriptions*.

Next Steps

If you're unable to reset the remote subscription, you may need to clear any outstanding remote subscription exceptions first.

Related Information

Clear Remote Subscription Exceptions [page 244]

Manage Remote Subscriptions [page 164]

7.2.11 Clear Remote Subscription Exceptions

Replication may be stopped due to remote subscription exceptions. For example, a remote subscription exception may be generated when a primary key violation, constraint violation, or null primary key occurs.

Remote subscription exceptions are reported in multiple ways.

- Remote subscription exceptions appear in the task execution log.
- E-mail alerts may be sent to the administrator.
- Exceptions are displayed in monitoring, or when you query the remote subscription exception public view.

After correcting the root cause of the remote subscription exception, you can clear the exceptions with a SQL statement:

```
PROCESS REMOTE SUBSCRIPTION EXCEPTION <subscription_id> IGNORE
```

7.3 Recovering from Crashes and Unplanned System Outages

Resume replication and ensure data consistency when you experience a crash or unplanned system outage.

Recover from an Index Server Crash [page 245]
Verify your data consistency when you experience an SAP HANA index server crash.

Recover from a Data Provisioning Server Crash [page 245]
Restart replication when the Data Provisioning Server crashes.

Recover from a Data Provisioning Agent JVM Crash [page 246]
The Data Provisioning Agent may crash if the Java virtual machine is configured with insufficient available memory.

Recover from an Unplanned Source Database Outage [page 247]
Replication may stop if there's an unplanned source database outage due to a network or hardware issue.

Recover from an SAP ASE Adapter Factory Crash [page 248]
The `dpadapterfactory` process used in the SAP ASE and SAP ECC ASE adapters restarts automatically if it's stopped or crashes.

Related Information

7.3.1 Recover from an Index Server Crash

Verify your data consistency when you experience an SAP HANA index server crash.

Procedure

1. After SAP HANA restarts, verify that all SAP HANA processes show an active GREEN status.
2. Compare source and target table row counts to verify data consistency.

7.3.2 Recover from a Data Provisioning Server Crash

Restart replication when the Data Provisioning Server crashes.

Context

When the Data Provisioning Server crashes, an error may be indicated in the `usr/sap/<sid>HDB<instance>/<hana_machine_name>/trace/dpserver_alert_<hana_machine_name>.trc` log file.

For example:

```
Instance LR1/00, OS Linux lsdub0007 3.0.101-80-default #1 SMP Fri Jul 15
14:30:41 UTC 2016 (eb2ba81) x86_64
----> Register Dump <----
  rax: 0x00007f638b068a90  rbx: 0x00007f5fa74a8010
  rcx: 0x0000000000000000  rdx: 0x0000000000000056
......
  xmm[14]: 0x00000000.3fffffff.3fffffff.00000000
  xmm[15]: 0x3fffffff.00000000.03020100.07060504
NOTE: full crash dump will be written to /usr/sap/LR1/HDB00/lsdub0007/trace/
dpserver_lsdub0007.30011.crashdump.20161102-143544.019227.trc
Call stack of crashing context:
 1: 0x0000000000000000 in <no symbol>+0x0 (<unknown>)
```

Recover from the crash by restarting replication and processing any remote subscription exceptions.

Procedure

1. Suspend all remote sources.
2. Restart the Data Provisioning Server.
3. Restart the SAP HANA system.
4. Resume all remote sources.

5. Process any remote subscription exceptions remaining after change data capture has resumed.

Next Steps

In a multiple container SAP HANA configuration, the Data Provisioning Server is managed individually in each tenant database. Repeat the recovery process for each tenant database.

Related Information

Processing Remote Source or Remote Subscription Exceptions [page 165]
Suspend and Resume Remote Sources [page 162]

7.3.3 Recover from a Data Provisioning Agent JVM Crash

The Data Provisioning Agent may crash if the Java virtual machine is configured with insufficient available memory.

Context

An error may be indicated in the `<DPAgent_root>/log/framework.trc` log file.

For example:

```
[ERROR] OracleRepAgentWrapper$2.run  - LogReader is not in REPLICATE state.
State: REPLICATION DOWN.
[ERROR] ReceiverImpl.sendError  - LogReader is in ERROR state [GC overhead limit
exceeded]. Check LogReader log for details.
[ERROR] ReceiverImpl.sendError  - Receiver is closed due to adapter error.Please
check REMOTE_SUBSCRIPTION_EXCEPTIONS table
```

Recover by adjusting the agent's maximum available memory and restarting the agent.

Procedure

1. Check the available memory on the Data Provisioning Agent host machine.

   For example:

   ```
   # free -m
                total       used       free     shared    buffers     cached
   Mem:        258313     196343      61970          0        589      53022
   ```

```
-/+ buffers/cache:     142731     115582
Swap:         2047          0       2047
```

2. Stop the Data Provisioning Agent service.
3. Adjust the agent maximum available memory in the agent runtime options.
4. Restart the Data Provisioning Agent service.

Related Information

Manage the Agent Service
Agent Runtime Options

7.3.4 Recover from an Unplanned Source Database Outage

Replication may stop if there's an unplanned source database outage due to a network or hardware issue.

> **Note**
>
> On Oracle, the Oracle Log Reader adapter automatically reconnects to the source database when available and resumes replication without any additional actions.

Context

A source database outage may be indicated by multiple error types in the logs.

For example:

```
Could not connect to <jdbc:oracle:thin:@host:1675:DDCSTD>: The Network Adapter
could not establish the connection
```

```
java.lang.Exception: Log scanner <LogMinerScanner_1> stopped because of error:
No more data to read from socket
```

Recover by processing remote subscription exceptions and resuming replication on the remote source.

Procedure

1. Suspend the remote source until the outage is resolved.
2. Process any remote subscription exceptions.
3. Resume replication on the remote source.

Related Information

Processing Remote Source or Remote Subscription Exceptions [page 165]
Suspend and Resume Remote Sources [page 162]

7.3.5 Recover from an SAP ASE Adapter Factory Crash

The `dpadapterfactory` process used in the SAP ASE and SAP ECC ASE adapters restarts automatically if it's stopped or crashes.

Context

An adapter factory crash may be indicated in the `<DPAgent_root>/log/framework.trc` log file.

For example:

```
[ERROR] com.sap.hana.dp.adapter.framework.cpp.CppAdapterManager.read[236]    -
Socket closed
com.sap.hana.dp.adapter.sdk.AdapterException: Socket closed
at
com.sap.hana.dp.adapter.framework.socket.SocketConnector.read(SocketConnector.jav
a:103)
at
com.sap.hana.dp.adapter.framework.cpp.CppAdapterManager.read(CppAdapterManager.ja
va:232)
```

Additional errors may be indicated in the `<DPAgent_root>/log/dpsadapterfactory_<##########>.txt` log file.

For example:

```
e Basis  FaultProtectionImpl.cpp(01610) + Instance (none)/(none), OS Linux
rqahana3 3.0.101-0.46-default #1 SMP Wed Dec 17 11:04:10 UTC 2014 (8356111)
x86_64
e Basis  FaultProtectionImpl.cpp(01610) + ----> Register Dump <----
e Basis  FaultProtectionImpl.cpp(01610) +   rax: 0xfffffffffffffffc  rbx:
0x00007f54e1adaa20
e Basis  FaultProtectionImpl.cpp(01610) +   rcx: 0xffffffffffffffff  rdx:
0x0000000000000000
……
Helper.cpp(00514) : Using 'x64_64 ABI unwind' for stack tracing
e Basis  FaultProtectionImpl.cpp(01610) + NOTE: full crash
dump will be written to /rqahana3_work2/songp/DPAgent_1.3.0_5030/log/
dpsadapterfactory.crashdump.20161020-233253.081342.trc
```

A full crash dump is written to `<DPAgent_root>/log/dpsadapterfactory.crashdump.<timestamp>.trc`

Resolve the errors by processing any remote exceptions.

Related Information

Processing Remote Source or Remote Subscription Exceptions [page 165]

7.4 Troubleshooting Data Provisioning Agent Issues

This section describes error situations related to the Data Provisioning Agent and their solutions.

Data Provisioning Agent Log Files and Scripts [page 249]
: Various log files are available to monitor and troubleshoot the Data Provisioning Agent.

Clean an Agent Started by the Root User [page 250]
: If the Data Provisioning Agent is accidentally started by the `root` user, you should clean up the agent installation.

Agent JVM Out of Memory [page 251]
: If the Java Virtual Machine on the Data Provisioning Agent is running out of memory, you may need to adjust configuration parameters on the Agent or your adapters.

Adapter Prefetch Times Out [page 252]
: When task execution fails due to an adapter timeout, you may need to adjust the adapter prefetch timeout parameter.

Uninstalled Agent Reports Alerts or Exceptions [page 252]
: After uninstalling the Data Provisioning Agent and installing a new version, alerts and exceptions may be reported for the previous installation.

Create an Agent System Dump [page 253]
: Use the command-line agent configuration tool to generate system dumps when troubleshooting the Data Provisioning Agent.

Resolve Agent Parameters that Exceed JVM Capabilities [page 254]
: Replication tasks may fail during the initial load if the fetch size exceeds the capabilities of the Data Provisioning Agent JVM.

Related Information

7.4.1 Data Provisioning Agent Log Files and Scripts

Various log files are available to monitor and troubleshoot the Data Provisioning Agent.

The following log files are available:

Log File Name and Location	Description
<DPAgent_root>/log/dpagent_service_eventlog_<date>.log	Data Provisioning Agent event log
<DPAgent_root>/log/framework_alert.trc	Data Provisioning Agent adapter framework log. Use this file to monitor data provisioning agent statistics.
<DPAgent_root>/log/framework.trc	Data Provisioning Agent adapter framework trace log. Use this file to trace and debug data provisioning agent issues.
<DPAgent_root>/LogReader/<instance>/log/<instance>.log	Instance log for log reader-based adapters
<DPAgent_root>/LogReader/admin_logs/admin<timestamp>.log	Administrative log for log reader-based adapters

Additionally, scripts for performing source database initialization and cleanup operations can be found at <DPAgent_root>/LogReader/scripts.

7.4.2 Clean an Agent Started by the Root User

If the Data Provisioning Agent is accidentally started by the root user, you should clean up the agent installation. In a normal configuration, the Data Provisioning Agent should be started only by a user with the same rights and permissions as the installation owner defined during the agent installation.

Context

If the agent is started by the root user, additional files are created in the installation location. The agent can't access these additional files because root is their owner, and they should be removed.

> **Note**
>
> This applies only if the agent was started by the user named root, and not other users that may belong to the root group or have similar permissions.

Procedure

1. Navigate to the configuration directory in the agent installation location.

   For example, /usr/sap/dataprovagent/configuration
2. Remove the following directories:
   - com.sap.hana.dp.adapterframework
   - org.eclipse.core.runtime

- `org.eclipse.osgi`

> ⚠ Caution
>
> Don't remove the `config.ini` file or `org.eclipse.equinox.simpleconfigurator` directory.

3. Remove the `log` directory and `secure_storage` file.

   For example, `/usr/sap/dataprovagent/log` and `/usr/sap/dataprovagent/secure_storage`.

> ⓘ Note
>
> Depending on the permissions, you may require sudo access to remove the `secure_storage` file.
>
> For example, `sudo rm -rf secure_storage`.

7.4.3 Agent JVM Out of Memory

If the Java Virtual Machine on the Data Provisioning Agent is running out of memory, you may need to adjust configuration parameters on the Agent or your adapters.

Oracle Log Reader Adapters

For Oracle log reader adapters, there are multiple possible solutions.

- Check the *Queue size of parallel scan tasks* parameter.
  The default value is 0, which allows for an unlimited queue size. To find the maximum queue size that results in acceptable memory use, set the queue size to 10,000 and increase in increments of 50,000 or 100,000.
- Check the *Parallel scan SCN range* parameter.
  Decrease the value as necessary.
- Check the *Number of parallel scanners* parameter.
  Decrease the value from 4 to 2.

Setting Virtual Memory

For general memory issues, you can increase the JVM memory usage.

In `dpagent.ini`, change the memory allocation parameter, and restart the Agent.

For example, to increase the memory to 8 GB, enter `-Xmx8096m`.

Memory Dump to a File Can Cause a Disk Full Error

The `dpagent.ini` file contains a parameter (`-XX:+HeapDumpOnOutOfMemoryError`) that allows you to perform a full memory dump, and the size of this dump is based on the `-xmx` parameter, which is also in the `dpagent.ini` file. For example, if you set the `-xmx` parameter to 64 GB, the file created during the dump will be 64 GB.

You can remove the `-XX:+HeapDumpOnOutOfMemoryError` parameter, or you can reduce the amount of space that the dump file will take up, to help avoid any "disk full" errors.

However, you should add the line back in, if needed, when you are attempting to debug out-of-memory errors of the Data Provisioning Agent. You would also need to ensure that you have enough disk space for the dump file.

7.4.4 Adapter Prefetch Times Out

When task execution fails due to an adapter timeout, you may need to adjust the adapter prefetch timeout parameter.

In the Index Server trace log, you may notice an error similar to the following:

```
DPAdapterAccess.cpp(01825) : DPAdapterAccess::Fetch: failed with error: Prefetch
timed out.
```

When this type of error appears, you can adjust the value of the adapter prefetch timeout parameter.

From a SQL console:

```
ALTER SYSTEM ALTER CONFIGURATION ('dpserver.ini','SYSTEM')
SET ('framework','prefetchTimeout')='<seconds>' WITH RECONFIGURE;
```

For example, you might change the value to **`600`** to set the timeout to 10 minutes.

7.4.5 Uninstalled Agent Reports Alerts or Exceptions

After uninstalling the Data Provisioning Agent and installing a new version, alerts and exceptions may be reported for the previous installation.

If the agent is uninstalled, but associated remote sources and the agent registration aren't dropped, any replication tasks still using remote sources on the previous agent may report alerts and exceptions.

Solution

Drop outdated remote sources and the agent registration referring to the old installation.

To suspend and drop a remote source:

```
ALTER REMOTE SOURCE "<remote_source_name>" SUSPEND CAPTURE;
DROP REMOTE SOURCE "<remote_source_name>" CASCADE;
```

To drop an agent registration:

```
DROP AGENT "<agent_name>" CASCADE;
```

If you don't know the old agent name, you can find it using `SELECT * FROM sys.m_agents`.

7.4.6 Create an Agent System Dump

Use the command-line agent configuration tool to generate system dumps when troubleshooting the Data Provisioning Agent.

Context

By default, the configuration tool generates system dumps that include the following information:

- Log reader, framework, and OSGi logs and traces
- Information about running Java and Data Provisioning Agent processes
- Information about JVM, threads, OSGi, and adapters, if the agent connection is available

> **Note**
>
> To export a system dump, the Data Provisioning Agent must be started and running.

Procedure

1. At the command line, navigate to `<DPAgent_root>/bin`.
2. Execute the command using the `--createFullSystemDump` parameter.
   - On Windows, **`agentcli.bat --createFullSystemDump [<additional_parameters>]`**
   - On Linux, **`./agentcli.sh --createFullSystemDump [<additional_parameters>]`**

Table 42: Supported Parameters

Parameter	Description
`--snapshotName <name>`	Specifies the name for the exported dump. By default, the exported filename is `<hostname>_<time>_<name>`. If no snapshot name is specified, "dpagent" is used as the name.
`--runtimeOnly`	Restricts the exported dump to runtime information only. The exported dump doesn't include a copy of the logs or other folders.

Related Information

Manage the Agent Service

7.4.7 Resolve Agent Parameters that Exceed JVM Capabilities

Replication tasks may fail during the initial load if the fetch size exceeds the capabilities of the Data Provisioning Agent JVM.

Context

An error may be indicated during the initial load.

For example:

```
Could not execute 'SELECT * FROM "SYSTEM"."V_ZMTEST_DPSIMPLE"' in 10:00.197
minutes .
SAP DBTech JDBC: [403]: internal error: Error opening the cursor for the remote
database Prefetch timed out. for query "SELECT "V_ZMTEST_DPSIMPLE"."COL1",
"V_ZMTEST_DPSIMPLE"."COL2", "V_ZMTEST_DPSIMPLE"."COL3",
"V_ZMTEST_DPSIMPLE"."COL4", "V_ZMTEST_DPSIMPLE"."COL5",
"V_ZMTEST_DPSIMPLE"."COL6", "V_ZMTEST_DPSIMPLE"."COL7" FROM "QARUSER.DPTEST01"
"V_ZMTEST_DPSIMPLE" "
```

In addition, an error may be indicated in the `<DPAgent_root>/log/framework.trc` log file.

For example:

```
[FATAL] DPFramework | WorkerThread.processRequest  - Java heap space (failed
to allocate 40000016 bytes) (max heap: 256 MB)[FATAL] DPFramework |
WorkerThread.processRequest  - Java heap space (failed to allocate 40000016
bytes) (max heap: 256 MB)
```

Resolve the errors by reducing the row fetch size and increasing the JVM maximum size.

Procedure

1. Drop all replication tasks.
2. Increase the row fetch size in the Agent Preferences.
3. Increase the maximum virtual machine size in the Agent Runtime Options.
4. Restart the Data Provisioning Agent.
5. Process any remote subscription exceptions.

Related Information

Processing Remote Source or Remote Subscription Exceptions [page 165]
Manage the Agent Service
Agent Configuration Parameters
Agent Runtime Options

7.5 Troubleshooting Other Issues

This section describes various issues unrelated to replication failures or the Data Provisioning Agent and their solutions.

Activate Additional Trace Logging for the Data Provisioning Server [page 256]
The trace logs contain detailed information about actions in the Data Provisioning server.

Resolve a Source and Target Data Mismatch [page 258]
If data in the source and target tables of a replication task is mismatched, fix the error by re-executing the replication task.

Configuring the Operation Cache [page 258]
You can improve performance by using an operation cache in some script servers.

Ensure Workload Management and Resource Consumption [page 260]
Ensure that in circumstances of limited memory or CPU resources that processes and system resources remain responsive.

Related Information

7.5.1 Activate Additional Trace Logging for the Data Provisioning Server

The trace logs contain detailed information about actions in the Data Provisioning server.When you're troubleshooting issues, you can gain additional insight about the root cause by increasing the level of detail in the logs.

Prerequisites

To configure traces, you must have the system privilege TRACE ADMIN.

Procedure

1. Log on to SAP HANA studio with a user that has system privilege TRACE ADMIN.
2. In the Administration editor, choose the *Trace Configuration* tab.
3. Choose *Edit Configuration* under "Database Trace".
4. Filter the component list for **dp***.
5. Set the *System Trace Level* to **DEBUG** or for any relevant components under the *DPSERVER* and *INDEXSERVER* nodes.

Scenario	Traces

Debugging Initial Load	• *DPSERVER*
	  • `dpframework`
	  Trace information for message handling between the index server, Data Provisioning Server, and Data Provisioning Agent.
	  • `dpframeworkprefetch`
	  Prefetch-specific trace information.
	  • `dpframeworkmessagebody`
	  Trace information useful for diagnosing incorrect numbers of rows or suspected data corruption.
	  Note
	  This option can generate a large amount of data in the trace log.
	• *INDEXSERVER*
	  • `dpadaptermanager`
	  Trace information for initial smart data access requests.
	• *XSENGINE*
	  • `dpadaptermanager`
	  Trace information for initial smart data access requests from the SAP HANA Web-based Development Workbench.

Scenario	Traces

Debugging Real-time Replication	• *DPSERVER*
	  • `dpframework`
	  Trace information for message handling between the index server, Data Provisioning Server, and Data Provisioning Agent.
	  • `dpreceiver`
	  Trace information for message receiver, including the transfer of row data received by the framework.
	  • `dpdistributor`
	  Trace information for the message distributor.
	  • `dpapplier`
	  Trace information for the message applier.
	  • `dpserver`
	  Trace information for communication between the Data Provisioning Server and the index server.
	  • `dpremotesubscriptionmanager`
	  Trace information for remote subscription runtime details on the Data Provisioning Server side.
	  • `dpframeworkmessagebody`
	  Trace information useful for diagnosing incorrect numbers of rows or suspected data corruption.
	  Note
	  This option can generate a large amount of data in the trace log.
	• *INDEXSERVER*
	  • `dpdistributor`
	  Trace information for the message distributor.
	  • `dpapplier`
	  Trace information for the message applier.
	  • `dpremotesubscriptionmanager`
	  Trace information for remote subscription runtime details on the index server side.

Tip

To ensure that applicable trace information is captured at the time of the error and not overwritten by log wrapping, you can increase the maximum log size while capturing the traces. In the *GLOBAL* node, increase the value of the *maxfilesize* parameter.

6. Click *Finish*.

Results

Additional debug information for the modified components is added to the trace log.

→ Tip

You can also modify the trace log levels by issuing commands from a SQL console and manually specifying the component to adjust.

For example, to set the Data Provisioning Server *dpframework* trace to **DEBUG**:

```
ALTER SYSTEM ALTER CONFIGURATION ('dpserver.ini','SYSTEM') SET
('trace','dpframework') = 'DEBUG' WITH RECONFIGURE;
```

7.5.2 Resolve a Source and Target Data Mismatch

If data in the source and target tables of a replication task is mismatched, fix the error by re-executing the replication task.

7.5.3 Configuring the Operation Cache

You can improve performance by using an operation cache in some script servers.

The operation cache holds operation instances for Global Address Cleanse, Universal Data Cleanse, Geocode, and Type Identifier (TID), which are initialized and ready for use during task plan execution. This improves performance by avoiding the process of task plan operation initialization/create and deletion, and allows the reuse of the cached instances both inside a single plan and across plans.

Having more instances cached improves performance, but those additional cached instances consume more memory.

Operation cache instances are type-specific and are set in the file `scriptserver.ini`.

You can use the following monitoring views to verify the cached operations, usage count, and so on.

- Select * from sys.m_caches;
- Select * from sys.m_cache_entries;

The operation cache can be configured in SAP HANA studio by editing the file `scriptserver.ini`.

Enabling Operation Cache

To start operation cache instances, you must set the *enable_adapter_operation_cache* parameter to *yes*. The following sample SQL statement sets the parameter to *yes*.

```
ALTER SYSTEM ALTER CONFIGURATION ('scriptserver.ini','SYSTEM') SET
('adapter_operation_cache', 'enable_adapter_operation_cache') = 'yes' WITH
RECONFIGURE;
```

Changing the Operation Cache Default Settings

You can turn off the cache for each node, or change the default number of instances. The following sample SQL statement changes the number of instances.

```
ALTER SYSTEM ALTER CONFIGURATION ('scriptserver.ini', 'SYSTEM') SET
('adapter_operation_cache', 'gac')='30';
```

By default, the operation cache is enabled for all the supported types of operations (which is recommended) and has the following number of instances.

Table 43: Operation Cache Default Settings

Option	Number of instances default setting.
Geocode	10
Global Address Cleanse (in the Cleanse node)	20
Universal Data Cleanse (in the Cleanse node)	60
Type Identification	20

Ensuring That Real-Time Jobs Have Priority

The operation cache is used for both batch jobs and real-time jobs. Batch jobs can exhaust the operation cache, leaving insufficient resources to optimize the running of real-time jobs. If you run real-time jobs, use these settings to ensure that a dedicated number of operation cache instances are available for real-time tasks. By default, the number of instances made available to real-time tasks is half the total number of instances for each option.

Table 44: Real-Time Default Settings

Option	Number of instances default setting
Geocode	5
Global Address Cleanse (in the Cleanse node)	10
Universal Data Cleanse (in the Cleanse node)	30
Type Identification	0

Caching Performance with Global Address Cleanse

Within the Cleanse node, you can configure two options that are relevant to Global Address Cleanse. When caching is enabled, for better performance set these options as follows:

Table 45: Best Performance Global Address Cleanse Options

Option	Recommended setting
Country Identification Mode	assign

Option	Recommended setting
Default Country	NONE

7.5.4 Ensure Workload Management and Resource Consumption

Ensure that in circumstances of limited memory or CPU resources that processes and system resources remain responsive.

Information on the specifics and procedures of workload management is found in the *SAP HANA Administration Guide*.

SAP HANA smart data integration takes advantage of this framework and allows you to better handle circumstances of limited resources. Workload management in SAP HANA allows you to optimize for your system. This framework also works within the limited memory or CPU resources, as you define them in the workload class and mapping.

For example, if the workload class sets “STATEMENT THREAD LIMIT = 5”, then SAP HANA creates up to five instances per node or operation in parallel during the task plan execution.

If the workload class sets “STATEMENT MEMORY LIMIT = 2GB”, but any of the nodes or operation in the task plan require more than 2 GB of memory, then the task would fail with an error “[MEMORY_LIMIT_VIOLATION] Information about current memory composite-limit violation”.

Consider these options and constraints to create the best possible performance.

Related Information

Administration Information Map (SAP HANA Administration Guide)

8 SQL and System Views Reference for Smart Data Integration and Smart Data Quality

This section contains information about SQL syntax and system views that can be used in SAP HANA smart data integration and SAP HANA smart data quality.

For complete information about all SQL statements and system views for SAP HANA and other SAP HANA contexts, see the *SAP HANA SQL and System Views Reference*.

For information about the capabilities available for your license and installation scenario,refer to the Feature Scope Description (FSD) for your specific SAP HANA version on the SAP HANA Platform page.

SQL Functions [page 261]
Documents the built-in SQL functions that are provided with SAP HANA smart data integration and SAP HANA smart data quality.

SQL Statements [page 262]
SAP HANA smart data integration and SAP HANA smart data quality support many SQL statements to allow you to do such tasks as create agents and adapters, administer your system, and so on.

System Views [page 302]
System views allow you to query for various information about the system state using SQL commands. The results appear as tables.

Related Information

SAP HANA SQL and System Views Reference

8.1 SQL Functions

Documents the built-in SQL functions that are provided with SAP HANA smart data integration and SAP HANA smart data quality.

SESSION_CONTEXT Function [Smart Data Integration] [page 262]
Returns the value of session_variable assigned to the current user.

Related Information

8.1.1 SESSION_CONTEXT Function [Smart Data Integration]

Returns the value of session_variable assigned to the current user.

SESSION_CONTEXT is available for use in other areas of SAP HANA, not only SAP HANA smart data integration. Refer to the SESSION_CONTEXT topic for complete information. The information below is specific to smart data integration functionality.

Syntax

```
SESSION_CONTEXT(<session_variable>)
```

Description

A predefined session variables that is set by the server and is read-only (cannot be SET or UNSET) is 'TASK_EXECUTION_ID'.

Related Information

SESSION_CONTEXT Function (Miscellaneous) (SAP HANA SQL and System Views Reference)

8.2 SQL Statements

SAP HANA smart data integration and SAP HANA smart data quality support many SQL statements to allow you to do such tasks as create agents and adapters, administer your system, and so on.

ALTER ADAPTER Statement [Smart Data Integration] [page 264]
The ALTER ADAPTER statement alters an adapter. Refer to CREATE ADAPTER for a description of the AT LOCATION clause.

ALTER AGENT Statement [Smart Data Integration] [page 265]
The ALTER AGENT statement changes an agent's host name and/or port and SSL property if it uses the TCP protocol. It can also assign an agent to an agent group.

ALTER REMOTE SOURCE Statement [Smart Data Integration] [page 267]
The ALTER REMOTE SOURCE statement modifies the configuration of an external data source connected to the SAP HANA database.

ALTER REMOTE SUBSCRIPTION Statement [Smart Data Integration] [page 271]
The ALTER REMOTE SUBSCRIPTION statement allows the QUEUE command to initiate real-time data processing, and the DISTRIBUTE command applies the changes.

CANCEL TASK Statement [Smart Data Integration] [page 272]
Cancels a task that was started with START TASK.

CREATE ADAPTER Statement [Smart Data Integration] [page 274]
The CREATE ADAPTER statement creates an adapter that is deployed at the specified location.

CREATE AGENT Statement [Smart Data Integration] [page 276]
The CREATE AGENT statement registers connection properties of an agent that is installed on another host.

CREATE AGENT GROUP Statement [Smart Data Integration] [page 278]
The CREATE AGENT GROUP statement creates an agent clustering group to which individual agents can be assigned.

CREATE AUDIT POLICY Statement [Smart Data Integration] [page 279]
The CREATE AUDIT POLICY statement creates a new audit policy, which can then be enabled and cause the specified audit actions to occur.

CREATE REMOTE SOURCE Statement [Smart Data Integration] [page 281]
The CREATE REMOTE SOURCE statement defines an external data source connected to SAP HANA database.

CREATE REMOTE SUBSCRIPTION Statement [Smart Data Integration] [page 282]
The CREATE REMOTE SUBSCRIPTION statement creates a remote subscription in SAP HANA to capture changes specified on the entire virtual table or part of a virtual table using a subquery.

CREATE VIRTUAL PROCEDURE Statement [Smart Data Integration] [page 288]
Creates a virtual procedure using the specified programming language that allows execution of the procedure body at the specified remote source.

DROP ADAPTER Statement [Smart Data Integration] [page 290]
The DROP ADAPTER statement removes an adapter from all locations.

DROP AGENT Statement [Smart Data Integration] [page 291]
The DROP AGENT statement removes an agent.

DROP AGENT GROUP Statement [Smart Data Integration] [page 293]
The DROP AGENT GROUP statement removes an agent clustering group.

DROP REMOTE SUBSCRIPTION Statement [Smart Data Integration] [page 294]
The DROP REMOTE SUBSCRIPTION statement drops an existing remote subscription.

GRANT Statement [Smart Data Integration] [page 295]
GRANT is used to grant privileges and structured privileges to users and roles. GRANT is also used to grant roles to users and other roles.

PROCESS REMOTE SUBSCRIPTION EXCEPTION Statement [Smart Data Integration] [page 297]
The PROCESS REMOTE SUBSCRIPTION EXCEPTION statement allows the user to indicate how an exception should be processed.

START TASK Statement [Smart Data Integration] [page 298]
Starts a task.

Related Information

SQL Reference for Additional SAP HANA Contexts (SAP HANA SQL and System Views Reference)

8.2.1 ALTER ADAPTER Statement [Smart Data Integration]

The ALTER ADAPTER statement alters an adapter. Refer to CREATE ADAPTER for a description of the AT LOCATION clause.

Syntax

```
ALTER ADAPTER <adapter_name> [PROPERTIES <properties>]
| {ADD | REMOVE} LOCATION {DPSERVER | AGENT <agent_name>}
| REFRESH AT LOCATION {DPSERVER | AGENT <agent_name>}
```

Syntax Elements

<adapter_name>

The name of the adapter to be altered.

```
<adapter_name> ::= <identifier>
```

<agent_name>

The agent name if the adapter is set up on the agent.

```
<agent_name> ::= <identifier>
```

<properties>

The optional properties of the adapter, such as display_name. If display_name is not specified, then adapter_name appears in the user interface.

```
<properties> ::= <string_literal>
```

Description

The ALTER ADAPTER statement alters an adapter. Refer to CREATE ADAPTER for a description of the AT LOCATION clause.

Permissions

Role: sap.hana.im.dp.monitor.roles::Operations

System privilege: ADAPTER ADMIN

Examples

Add or remove an existing adapter at agent or Data Provisioning Server

Create two agents and an adapter at the first agent:

```
CREATE AGENT TEST_AGENT_1 PROTOCOL 'TCP' HOST 'test_host1'
PORT 5050;
CREATE AGENT TEST_AGENT_2 PROTOCOL 'HTTP';
CREATE ADAPTER TEST_ADAPTER AT LOCATION AGENT TEST_AGENT_1;
```

Add an existing adapter TEST_ADAPTER to agent TEST_AGENT_2:

```
ALTER ADAPTER TEST_ADAPTER ADD LOCATION AGENT TEST_AGENT_2;
```

Remove an existing adapter TEST_ADAPTER from agent TEST_AGENT_2:

```
ALTER ADAPTER TEST_ADAPTER REMOVE LOCATION AGENT
TEST_AGENT_2;
```

Add an existing adapter TEST_ADAPTER at the Data Provisioning Server:

```
ALTER ADAPTER TEST_ADAPTER ADD LOCATION DPSERVER;
```

Remove an existing adapter TEST_ADAPTER at Data Provisioning Server:

```
ALTER ADAPTER TEST_ADAPTER REMOVE LOCATION DPSERVER;
```

Refresh configuration and query optimization capabilities of an adapter

Read configuration and query optimization capabilities of an adapter from the adapter setup at the agent or Data Provisioning Server:

```
ALTER ADAPTER TEST_ADAPTER REFRESH AT LOCATION DPSERVER;
ALTER ADAPTER TEST_ADAPTER REFRESH AT LOCATION AGENT
TEST_AGENT_2;
```

Update display name property of an adapter

Change display name for an adapter to 'My Custom Adapter':

```
ALTER ADAPTER TEST_ADAPTER PROPERTIES 'display_name=My
Custom Adapter';
```

8.2.2 ALTER AGENT Statement [Smart Data Integration]

The ALTER AGENT statement changes an agent's host name and/or port and SSL property if it uses the TCP protocol. It can also assign an agent to an agent group.

Syntax

```
ALTER AGENT <agent_name>
HOST <agent_hostname> [ PORT <agent_port_number> ] [ { ENABLE | DISABLE } SSL ]
```

```
| PORT <agent_port_number> [ {ENABLE | DISABLE} SSL ]
| [ {ENABLE | DISABLE} SSL ]
| { SET | UNSET } AGENT GROUP <agent_group_name>
```

Syntax Elements

<agent_name>

The name of the agent to be altered.

```
<agent_name> ::= <identifier>
```

<agent_hostname>

The name of the agent host.

```
<agent_hostname> ::= <string_literal>
```

<agent_port_number>

Specifies whether the agent's TCP listener on the specified port uses SSL.

```
<agent_port_number> ::= <integer_literal>  {ENABLE | DISABLE}
SSL
```

<agent_group_name>

The name of the agent clustering group to which the agent should be attached.

```
<agent_group_name> ::= <identifier>
```

Description

The ALTER AGENT statement changes an agent's host name and/or port if it uses the TCP protocol. It can also assign an agent to an agent group.

Permissions

Role: sap.hana.im.dp.monitor.roles::Operations

Application privilege: sap.hana.im.dp.monitor::AlterAgent

System privilege: AGENT ADMIN

Examples

- Alter TEST_AGENT's hostname to test_host and port to 5051, if it uses TCP protocol:

```
ALTER AGENT TEST_AGENT HOST 'test_host' PORT 5051;
```

- Alter TEST_AGENT's hostname to test_host, if it uses TCP protocol:

```
ALTER AGENT TEST_AGENT HOST 'test_host';
```

- Alter TEST_AGENT's port to 5051, if it uses TCP protocol:

```
ALTER AGENT TEST_AGENT PORT 5051;
```

- Assign TEST_AGENT to agent group TEST_GROUP:

```
ALTER AGENT TEST_AGENT SET AGENT GROUP TEST_GROUP;
```

- Remove TEST_AGENT from agent group TEST_GROUP:

```
ALTER AGENT TEST_AGENT UNSET AGENT GROUP TEST_GROUP;
```

8.2.3 ALTER REMOTE SOURCE Statement [Smart Data Integration]

The ALTER REMOTE SOURCE statement modifies the configuration of an external data source connected to the SAP HANA database.

The ALTER REMOTE SOURCE SQL statement is available for use in other areas of SAP HANA, not only SAP HANA smart data integration. Refer to the ALTER REMOTE SOURCE topic for complete information. This information is specific to smart data integration functionality.

Syntax

```
ALTER REMOTE SOURCE <remote_source_name> <adapter_clause> [<credential_clause>]
| { SUSPEND | RESUME } { CAPTURE | DISTRIBUTION }
| { CLEAR OBJECTS | REFRESH OBJECTS | CANCEL REFRESH OBJECTS }
| START LATENCY MONITORING <latency_ticket_name> [ INTERVAL
<interval_in_seconds> ]
| STOP LATENCY MONITORING <latency_ticket_name>
| CLEAR LATENCY HISTORY [ <latency_ticket_name> ]
```

Syntax Elements

Syntax elements specific to smart data integration are described as follows. For information about syntax elements that aren't specific to smart data integration, refer to the ALTER REMOTE SOURCE topic.

<adapter_clause>

Adapter configuration.

```
<adapter_clause> ::= [ADAPTER <adapter_name>
[AT LOCATION { DPSERVER | AGENT <agent_name> | AGENT GROUP
<agent_group_name>}] <configuration_clause>]
```

```
<agent_name> ::= <identifier>
<agent_group_name> ::= <identifier>
```

```
<configuration_clause> ::= CONFIGURATION
'<configuration_xml_string>'
```

The <configuration_xml_string> is the XML-formatted configuration string for the remote source.

Refer to CREATE ADAPTER for a description of the AT LOCATION clause.

{ SUSPEND | RESUME } { CAPTURE | DISTRIBUTION }

ALTER REMOTE SOURCE SUSPEND CAPTURE	Suspends the adapter and agent from reading any more changes from source system. This is helpful when the source system or SAP HANA is preparing for planned maintenance or an upgrade.
ALTER REMOTE SOURCE **<remote_source_name>** RESUME CAPTURE	Resumes the suspended adapter to read changed data from source system.
ALTER REMOTE SOURCE **<remote_source_name>** SUSPEND DISTRIBUTION	Suspends the application of real-time changes in SAP HANA tables but collects changed data from the source system.
ALTER REMOTE SOURCE **<remote_source_name>** RESUME DISTRIBUTION	Resumes applying real-time changes in SAP HANA tables.

{ CLEAR OBJECTS | REFRESH OBJECTS | CANCEL REFRESH OBJECTS }

ALTER REMOTE SOURCE **<remote_source_name>** CLEAR OBJECTS	Clears all the data received from the adapter for this remote source from HANA tables.
ALTER REMOTE SOURCE **<remote_source_name>** REFRESH OBJECTS	Starts building HANA dictionary tables that contain remote source objects.
ALTER REMOTE SOURCE **<remote_source_name>** CANCEL REFRESH OBJECTS	Cancels the long-running REFRESH background operation. This stops fetching records from the adapter but keeps the data received so far form the remote source on HANA tables.

ALTER REMOTE SOURCE **<remote_source_name>** START LATENCY MONITORING **<ticket_name>**

Starts the collection of latency statistics one time or at regular intervals. The user specifies a target latency ticket in the monitoring view.

ALTER REMOTE SOURCE `<remote_source_name>` STOP LATENCY MONITORING `<ticket_name>`

Stops the collection of latency statistics into the given latency ticket.

ALTER REMOTE SOURCE `<remote_source_name>` CLEAR LATENCY HISTORY

Clears the latency statistics (for either one latency ticket, or for the whole remote source, from the monitoring view.

Description

The ALTER REMOTE SOURCE statement modifies the configuration of an external data source connected to the SAP HANA database. Only database users with the object privilege ALTER for remote sources or the system privilege DATA ADMIN may alter remote sources.

> **Note**
>
> You may not change a user name while a remote source is suspended.
>
> > **Note**
> >
> > When altering `CloudDataIntegrationAdapter` remote sources, SAP recommends using WebIDE (or other such SAP interfaces like Database explorer.) If using the ALTER REMOTE SOURCE command, refer to 3474959 for an SAC CDI example.

Permissions

This statement requires the ALTER object privilege on the remote source.

Examples

```
ALTER REMOTE SOURCE "odata_nw" ADAPTER "ODataAdapter"
AT LOCATION DPSERVER
CONFIGURATION '<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ConnectionProperties name="connection_properties">
<PropertyEntry name="URL">http://services.odata.org/Northwind/Northwind.svc/</
PropertyEntry>
<PropertyEntry name="proxyserver">proxy</PropertyEntry>
<PropertyEntry name="proxyport">8080</PropertyEntry> <PropertyEntry
name="truststore"></PropertyEntry>
<PropertyEntry name="supportformatquery"></PropertyEntry>
</ConnectionProperties>' WITH CREDENTIAL TYPE 'PASSWORD'
USING '<CredentialEntry name="password"><user></user><password></password></
CredentialEntry>';
```

The configuration clause must be a structured XML string that defines the settings for the remote source. For example, the CONFIGURATION string in the following example configures a remote source for an Oracle database.

```
CONFIGURATION '<?xml version="1.0" encoding="UTF-8"?>
    <ConnectionProperties name="configurations">
        <PropertyGroup name="generic">
            <PropertyEntry name="instance_name">ora_inst</PropertyEntry>
            <PropertyEntry name="admin_port">12345</PropertyEntry>
            <PropertyEntry name="map_char_types_to_unicode">false</PropertyEntry>
        </PropertyGroup>
        <PropertyGroup name="database">
            <PropertyEntry name="cdb_enabled">false</PropertyEntry>
            <PropertyEntry name="pds_use_tnsnames">false</PropertyEntry>
            <PropertyEntry name="pds_host_name"><db_hostname></PropertyEntry>
            <PropertyEntry name="pds_port_number">1521</PropertyEntry>
            <PropertyEntry name="pds_database_name">ORCL</PropertyEntry>
            <PropertyEntry name="cdb_service_name"></PropertyEntry>
            <PropertyEntry name="pds_service_name"></PropertyEntry>
            <PropertyEntry name="pds_tns_filename"></PropertyEntry>
            <PropertyEntry name="pds_tns_connection"></PropertyEntry>
            <PropertyEntry name="cdb_tns_connection"></PropertyEntry>
            <PropertyEntry name="_pds_tns_connection_with_cdb_enabled"></
PropertyEntry>
            <PropertyEntry name="pds_byte_order"></PropertyEntry>
        </PropertyGroup>
        <PropertyGroup name="schema_alias_replacements">
            <PropertyEntry name="schema_alias"></PropertyEntry>
            <PropertyEntry name="schema_alias_replacement"></PropertyEntry>
        </PropertyGroup>
        <PropertyGroup name="security">
            <PropertyEntry name="pds_use_ssl">false</PropertyEntry>
            <PropertyEntry name="pds_ssl_sc_dn"></PropertyEntry>
            <PropertyEntry name="_enable_ssl_client_auth">false</PropertyEntry>
        </PropertyGroup>
        <PropertyGroup name="jdbc_flags">
            <PropertyEntry name="remarksReporting">false</PropertyEntry>
        </PropertyGroup>
        <PropertyGroup name="cdc">
            <PropertyGroup name="databaseconf">
                <PropertyEntry name="pdb_timezone_file"><timezone_file></
PropertyEntry>
                <PropertyEntry name="pdb_archive_path"></PropertyEntry>
                <PropertyEntry name="pdb_supplemental_logging_level">table</
PropertyEntry>
            </PropertyGroup>
            <PropertyGroup name="parallelscan">
                <PropertyEntry name="lr_parallel_scan">false</PropertyEntry>
                <PropertyEntry name="lr_parallel_scanner_count"></PropertyEntry>
                <PropertyEntry name="lr_parallel_scan_queue_size">></
PropertyEntry>
                <PropertyEntry name="lr_parallel_scan_range"></PropertyEntry>
            </PropertyGroup>
            <PropertyGroup name="logreader">
                <PropertyEntry name="skip_lr_errors">false</PropertyEntry>
                <PropertyEntry name="lr_max_op_queue_size">1000</PropertyEntry>
                <PropertyEntry name="lr_max_scan_queue_size">1000</PropertyEntry>
                <PropertyEntry name="lr_max_session_cache_size">1000</
PropertyEntry>
                <PropertyEntry name="scan_fetch_size">10</PropertyEntry>
                <PropertyEntry name="pdb_dflt_column_repl">true</PropertyEntry>
                <PropertyEntry name="pdb_ignore_unsupported_anydata">false</
PropertyEntry>
                <PropertyEntry name="pds_sql_connection_pool_size">15</
PropertyEntry>
                <PropertyEntry name="pds_retry_count">5</PropertyEntry>
                <PropertyEntry name="pds_retry_timeout">10</PropertyEntry>
```

```
            </PropertyGroup>
        </PropertyGroup>
</ConnectionProperties>'
```

Related Information

ALTER REMOTE SOURCE Statement (Access Control) (SAP HANA SQL and System Views Reference)

8.2.4 ALTER REMOTE SUBSCRIPTION Statement [Smart Data Integration]

The ALTER REMOTE SUBSCRIPTION statement allows the QUEUE command to initiate real-time data processing, and the DISTRIBUTE command applies the changes.

Syntax

```
ALTER REMOTE SUBSCRIPTION [<schema_name>.]<subscription_name>
{ QUEUE | DISTRIBUTE | RESET }
```

Syntax Elements

<subscription_name>

The name of the remote subscription.

```
<subscription_name> ::= <identifier>
```

Description

The ALTER REMOTE SUBSCRIPTION statement allows the QUEUE command to initiate real-time data processing, and the DISTRIBUTE command applies the changes. Typically, the initial load of data is preceded by QUEUE command. The DISTRIBUTE command is used when initial load completes. The RESET command can be used to reset the real-time process to start from the initial load again.

Permissions

This statement requires the ALTER object privilege on the remote source.

Example

Capture changes from a virtual table to an SAP HANA table.

```
CREATE AGENT TEST_AGENT PROTOCOL 'TCP' HOST 'test_host1' PORT 5050;
CREATE ADAPTER 'DB2ECCAdapter' AT LOCATION AGENT TEST_AGENT;
CREATE REMOTE SOURCE MYECC ADAPTER 'DB2ECCAdapter' CONFIGURATION
'<configuration_xml>' AT LOCATION AGENT TEST_AGENT;
CREATE VIRTUAL TABLE MARA_VT AT MYECC."<NULL>"."<NULL>".MARA;
CREATE COLUMN TABLE TGT_MARA LIKE MARA_VT;
CREATE REMOTE SUBSCRIPTION TEST_SUB ON MARA_VT TARGET TABLE TGT_MARA;
ALTER REMOTE SUBSCRIPTION TEST_SUB QUEUE;
```

Perform initial load of data using INSERT-SELECT or a TASK.

```
INSERT INTO TGT_MARA SELECT * FROM MARA_VT;
ALTER REMOTE SUBSCRIPTION TEST_SUB DISTRIBUTE;
```

Now insert or update a material record in ECC system and see it updated to TGT_MARA table in SAP HANA. Reset the real-time process and restart the load.

```
ALTER REMOTE SUBSCRIPTION TEST_SUB RESET;
ALTER REMOTE SUBSCRIPTION TEST_SUB QUEUE;
```

Perform initial load of data using INSERT-SELECT or a TASK.

```
INSERT INTO TGT_MARA SELECT * FROM MARA_VT;
ALTER REMOTE SUBSCRIPTION TEST_SUB DISTRIBUTE;
```

8.2.5 CANCEL TASK Statement [Smart Data Integration]

Cancels a task that was started with START TASK.

Syntax

```
CANCEL TASK <task_execution_id> [WAIT <wait_time_in_seconds>]
```

Syntax Elements

```
<task_execution_id> ::= <unsigned_integer>
```

Specifies the task execution ID to cancel. See the START TASK topic for more information about TASK_EXECUTION_ID.

```
 <wait_time_in_seconds> ::= <identifier>
```

Number of seconds to wait for the task to cancel before returning from the command.

Description

Cancels a task that was started with START TASK.

The default behavior is for the CANCEL TASK command to return after sending the cancel request. Optionally, a WAIT value can be specified where the command will wait for the task to actually cancel before returning. If the command has waited the specified amount of time, then the CANCEL TASK will error out with the error code 526 (request to cancel task was sent but task did not cancel before timeout was reached).

> ⓘ Note
>
> If the WAIT value is 0, the command returns immediately after sending the cancel request, as it would if no WAIT value were entered.

Permissions

The user that called START TASK can implicitly CANCEL; otherwise, the CATALOG READ and SESSION ADMIN roles are required.

Examples

Assuming that a TASK performTranslation was already started using START TASK and has a task execution ID of 255, it would be cancelled using the following commands. The behavior is the same for the following two cases:

```
CANCEL TASK 255;
```

```
CANCEL TASK 255 WAIT 0;
```

Assuming that a TASK performTranslation was already started using START TASK and has a task execution id of 256 and the user wants to wait up to 5 seconds for the command to cancel, it would be cancelled using the following command:

```
CANCEL TASK 256 WAIT 5;
```

If the task was able to cancel within 5 seconds, the CANCEL TASK will return as a success. If it didn't cancel within 5 seconds, then the return will be the error code 526.

SQL Script

You can call CANCEL TASK within the SQL Script CREATE PROCEDURE. Refer to the SAP HANA SQL Script Reference for complete details about CREATE PROCEDURE.

```
CREATE PROCEDURE "CANCEL_TASK"."CANCEL_MY_TASK"(in exec_id INT)
LANGUAGE SQLSCRIPT AS
BEGIN
     CANCEL TASK :exec_id;
END;
```

CANCEL TASK is not supported in:

- Table UDF
- Scalar UDF
- Trigger
- Read-only procedures

Related Information

START TASK Statement [Smart Data Integration] [page 298]

8.2.6 CREATE ADAPTER Statement [Smart Data Integration]

The CREATE ADAPTER statement creates an adapter that is deployed at the specified location.

Syntax

```
CREATE ADAPTER <adapter_name> [PROPERTIES <properties>] AT LOCATION
{DPSERVER | AGENT <agent_name>}
```

Syntax Elements

<adapter_name>

The name of the adapter to be created.

```
<adapter_name> ::= <identifier>
```

<agent_name>

The agent name if the adapter is set up on the agent.

```
<agent_name> ::= <identifier>
```

<properties>

The optional properties of the adapter, such as display_name. When display_name is not specified, then adapter_name displays in the user interface.

```
<properties> ::= <string_literal>
```

AT LOCATION DPSERVER

The adapter runs inside the Data Provisioning Server process in SAP HANA.

AT LOCATION

Specify an agent that is set up outside of SAP HANA for the adapter to run inside.

```
AT LOCATION AGENT <agent_name>
```

Description

The CREATE ADAPTER statement creates an adapter that is deployed at the specified location. The adapter must be set up on the location prior to running this statement. When the statement is executed, the Data Provisioning Server contacts the adapter to retrieve its configuration details such as connection properties and query optimization capabilities.

Permissions

Role: sap.hana.im.dp.monitor.roles::Operations

Application privilege: sap.hana.im.dp.monitor::AddLocationToAdapter

System privilege: ADAPTER ADMIN

Examples

Create an adapter at the Data Provisioning Server

Create an adapter TEST_ADAPTER running in the Data Provisioning Server.

```
CREATE ADAPTER TEST_ADAPTER AT LOCATION DPSERVER;
```

Create an adapter at a specified agent

Create an agent with name TEST_AGENT.

```
CREATE AGENT TEST_AGENT PROTOCOL 'TCP' HOST
'test_host' PORT 5050;
```

Create an adapter TEST_ADAPTER on agent TEST_AGENT.

```
CREATE ADAPTER TEST_ADAPTER AT LOCATION AGENT
TEST_AGENT;
```

8.2.7 CREATE AGENT Statement [Smart Data Integration]

The CREATE AGENT statement registers connection properties of an agent that is installed on another host.

Syntax

```
CREATE AGENT <agent_name> PROTOCOL {'HTTP' [HOST <agent_hostname> PORT
<agent_port_number>] | 'TCP' HOST <agent_hostname> PORT <agent_port_number>
[{ENABLE | DISABLE} SSL]} [AGENT GROUP <agent_group_name>]
```

Syntax Elements

<agent_name>

The name of the agent to be created and its protocol.

```
<agent_name> ::= <identifier>
```

PROTOCOL

The protocol for the agent.

HTTP

Agent uses HTTP protocol for communication with DP server. Use this protocol when the SAP HANA database is on the cloud.

```
PROTOCOL 'HTTP' [HOST <agent_hostname> PORT
<agent_port_number>]
```

> → Tip
>
> While HOST and PORT are not mandatory when you use HTTP, we recommend that you provide them so that it is easier to distinguish between different agents that are registered.

In the M_AGENTS monitoring view, the protocol is displayed as HTTP when the agent connects to SAP HANA using HTTP mode (when xsengine is available) or JDBC mode.

TCP

Agent uses TCP protocol and listens on the specified port to receive requests from DP server. Use this protocol when the SAP HANA database can connect to agent's TCP port.

```
PROTOCOL 'TCP' HOST <agent_hostname> PORT
<agent_port_number>
```

```
<agent_hostname> ::= <string_literal>
<agent_port_number> ::= <integer_literal>
```

DP server connects to the agent listening on the specified hostname and port. Use this protocol when the SAP HANA database is on-premise.

{ENABLE | DISABLE} SSL

Specifies if agent's TCP listener on the specified port uses SSL.

<agent_group_name>

The name of the agent clustering group to which the agent should belong.

```
<agent_group_name> ::= <identifier>
```

Description

The CREATE AGENT statement registers connection properties of an agent that is installed on another host. The DP server and agent use these connection properties when establishing communication channel.

Permissions

Role: sap.hana.im.dp.monitor.roles::Operations

Application privilege: sap.hana.im.dp.monitor::CreateAgent

System privilege: AGENT ADMIN

Examples

Create an agent with TCP protocol	Create an agent TEST_AGENT running on test_host and port 5050. `CREATE AGENT TEST_AGENT PROTOCOL 'TCP' HOST 'test_host' PORT 5050;`
Create an agent with HTTP protocol	Create an agent TEST_AGENT that uses HTTP. `CREATE AGENT TEST_AGENT PROTOCOL 'HTTP';`
Create an agent with HTTP protocol in an agent group	Create an agent TEST_AGENT that uses HTTP and belongs to agent clustering group TEST_GROUP. `CREATE AGENT TEST_AGENT PROTOCOL 'HTTP' AGENT GROUP TEST_GROUP;`

8.2.8 CREATE AGENT GROUP Statement [Smart Data Integration]

The CREATE AGENT GROUP statement creates an agent clustering group to which individual agents can be assigned.

Syntax

```
CREATE AGENT GROUP <agent_group_name>
```

Syntax Elements

<agent_group_name>

The name of the agent group to create.

```
<agent_group_name> ::= <identifier>
```

Description

The CREATE AGENT GROUP statement creates an agent clustering group to which individual agents can be assigned. An agent group can be used instead of a single agent to provide fail-over capabilities.

Permissions

Role: sap.hana.im.dp.monitor.roles::Operations

Application privilege: sap.hana.im.dp.monitor::CreateAgent

System privilege: AGENT ADMIN

Examples

Create an agent group named TEST_GROUP.

```
CREATE AGENT GROUP TEST_GROUP;
```

Related Information

ALTER AGENT Statement [Smart Data Integration] [page 265]
CREATE AGENT Statement [Smart Data Integration] [page 276]

8.2.9 CREATE AUDIT POLICY Statement [Smart Data Integration]

The CREATE AUDIT POLICY statement creates a new audit policy, which can then be enabled and cause the specified audit actions to occur.

The CREATE AUDIT POLICY SQL statement is available for use in other areas of SAP HANA, not only SAP HANA smart data integration. Refer to the CREATE AUDIT POLICY topic for complete information. The information below is specific to smart data integration functionality.

Syntax

Refer to the *SAP HANA SQL and System Views Reference* for complete information about CREATE AUDIT POLICY syntax.

Syntax Elements

Syntax elements specific to smart data integration are described below. For information about syntax elements that are not specific to smart data integration, refer to the *SAP HANA SQL and System Views Reference*.

```
<audit_action_name> ::= CREATE AGENT
    | ALTER AGENT
    | DROP AGENT
    | CREATE AGENT GROUP
    | DROP AGENT GROUP
    | CREATE ADAPTER
    | ALTER ADAPTER
    | DROP ADAPTER
    | CREATE REMOTE SUBSCRIPTION
    | ALTER REMOTE SUBSCRIPTION
    | DROP REMOTE SUBSCRIPTION
    | PROCESS REMOTE SUBSCRIPTION EXCEPTION
```

Audit Action Name	Group Number	Audit Operation
CREATE AGENT	17	Registering a Data Provisioning Agent
ALTER AGENT	17	Altering a Data Provisioning Agent's registration
DROP AGENT	17	Dropping a Data Provisioning Agent registration
CREATE ADAPTER	17	Registering a Data Provisioning Adapter
ALTER ADAPTER	17	Altering the registration of a Data Provisioning Adapter
DROP ADAPTER	17	Dropping the registration of a Data Provisioning Adapter
CREATE REMOTE SUBSCRIPTION	17	Creating a subscription to a remote source
ALTER REMOTE SUBSCRIPTION	17	Altering a subscription to a remote source
DROP REMOTE SUBSCRIPTION	17	Dropping a subscription to a remote source
PROCESS REMOTE SUBSCRIPTION EXCEPTION	17	Processing exceptions raised by a subscribed remote source

Description

The CREATE AUDIT POLICY statement creates a new audit policy. This audit policy can then be enabled and cause the auditing of the specified audit actions to occur.

Permissions

Only database users with the CATALOG READ, DATA ADMIN, or INIFILE ADMIN system privilege can view information in the M_INIFILE_CONTENTS view. For other database users, this view is empty. Users with the AUDIT ADMIN privilege can see audit-relevant parameters.

Related Information

CREATE AUDIT POLICY Statement (Access Control) (SAP HANA SQL and System Views Reference)

8.2.10 CREATE REMOTE SOURCE Statement [Smart Data Integration]

The CREATE REMOTE SOURCE statement defines an external data source connected to SAP HANA database.

The CREATE REMOTE SOURCE SQL statement is available for use in other areas of SAP HANA, not only SAP HANA smart data integration. Refer to the CREATE REMOTE SOURCE topic for complete information. The information below is specific to smart data integration functionality.

Syntax

Refer to the *SAP HANA SQL and System Views Reference* for complete information about CREATE REMOTE SOURCE syntax.

Syntax Elements

Syntax elements specific to smart data integration are described below. For information about syntax elements that are not specific to smart data integration, refer to the *SAP HANA SQL and System Views Reference*.

<adapter_clause>

Configures the adapter.

```
<adapter_clause> ::= ADAPTER <adapter_name>
     [AT LOCATION {DPSERVER | AGENT <agent_name> | AGENT GROUP
<agent_group_name>} ]
     CONFIGURATION <connection_info_string>
```

```
<agent_name> ::= <identifier>
<agent_group_name> ::= <identifier>
```

Refer to CREATE ADAPTER for description of AT LOCATION.

Description

The CREATE REMOTE SOURCE statement defines an external data source connected to SAP HANA database. Only database users having the system privilege CREATE SOURCE or DATA ADMIN are allowed to add a new remote source.

Permissions

This statement requires the CREATE SOURCE system privilege.

Related Information

CREATE REMOTE SOURCE Statement (Access Control) (SAP HANA SQL and System Views Reference)

8.2.11 CREATE REMOTE SUBSCRIPTION Statement [Smart Data Integration]

The CREATE REMOTE SUBSCRIPTION statement creates a remote subscription in SAP HANA to capture changes specified on the entire virtual table or part of a virtual table using a subquery.

Syntax

```
CREATE REMOTE SUBSCRIPTION [<schema_name>.]<subscription_name>
{
   {ON [<schema_name>.]<virtual_table_name> } |
   {AS (<subquery>)}
}
[ WITH SCHEMA CHANGES | WITH RESTRICTED SCHEMA CHANGES ]
{ TARGET TABLE <table_spec> <load_behavior> } |
{ TARGET TASK <task_spec> } |
{ TARGET PROCEDURE <proc_spec> }
```

Syntax Elements

<subscription_name>

The name of the remote subscription.

```
<subscription_name> ::= <identifier>
```

ON [<schema_name>.]<virtual_table_name>

See "Remote subscription for TARGET TASK or TARGET TABLE using ON Clause" below.

AS (<subquery>)

See "Remote subscription for TARGET TASK or TARGET TABLE using AS Clause" below.

[WITH SCHEMA CHANGES]

Include this clause to propagate source schema changes to the SAP HANA virtual table and remote subscription target table.

WITH SCHEMA CHANGES corresponds to the replication task options *Initial + realtime with structure* or *Realtime only with structure* and the flowgraph options *Real-time* and *with Schema Change*.

[WITH RESTRICTED SCHEMA CHANGES]

The restrict schema changes option is supported only on subscriptions where the target is a TARGET TABLE. It is not supported on subscription targets of TARGET TASK or PROCEDURE.

Restricted schema changes mode is intended to allow jobs dependent on the schema of the target table to continue running even when some schema changes have been made to the source table because the target table column names are unchanged. When possible, the target table data continues to be updated as the source table data changes.

Schema changes made to the source table that affect column names and number of columns are not propagated to the target table. Dropping or adding columns in the source table does not affect the target table.

Behavior of [WITH RESTRICTED SCHEMA CHANGES] ON [<schema_name>].<virtual_table_name>

- **Add Column**
  Columns may be added to the source table and will not be added to the target table. The target table continues to be updated, but as it is missing the newly-added columns, source table data for the added columns is not promulgated to the target table.
  The column is added to the virtual table.
- **Drop Column**
  Columns may be dropped from the source table and will not be dropped from the target table. The dropped column remains in the target table with its values preserved up to the point of the drop. After the drop, the dropped column still present in the target table is populated with null values as rows are added to the source table.
  The column is dropped from the virtual table.
- **Rename Column**
  Non-primary key columns may be renamed and will not be renamed in the target table. The originally-named column remains in the target table with its values preserved up to the time of the rename. After the renaming, the originally-named column still present in the target table is populated with null values as rows are added to the target table. Attempting to rename a column that is a primary key results in a subscription exception.

The column is renamed in the virtual table.

- **Alter Column**
  The column data type may be changed and will change the data type of the column in the target table. The new data type must be one supported by SAP HANA.
  The column data type is also changed in the virtual table.

Behavior of [WITH RESTRICTED SCHEMA CHANGES] with `<subquery>`

- **Add Column**
  Columns may be added to the source table and will not be added to the target table. The target table continues to be updated, but as it is missing the newly-added columns, source table data for the added columns is not promulgated to the target table.
  If the added column is part of the subquery, it is added to the virtual table.
- **Drop Column**
  Dropping columns from the source table results in a subscription error if the dropped columns are part of the subquery. Source table columns that are not part of the subscription subquery may be dropped.
  The column is not dropped from the virtual table because the subscription error prevents it.
- **Rename Column**
  Source table columns may be renamed, but the change is not promulgated to the target table. The subquery is updated with the new column name and the virtual table is changed to reflect the new name.
  With the subquery option, the originally-named target table column continues to be updated as rows are added to the source table, even though the column name differs between the subquery and the target table.
- **Alter Column**
  The column data type may be changed and will change the data type of the column in the target table. The new data type must be one supported by SAP HANA.
  The column data type is also changed in the virtual table.

`<table_spec>`

The table definition.

```
<table_spec> ::= [<schema_name>].<table_name>
```

`<load_behavior>`

```
[CHANGE TYPE COLUMN <column_name> CHANGE TIME COLUMN
<column_name> CHANGE SEQUENCE COLUMN <column_name> {INSERT |
UPSERT | ARCHIVE}]
```

CHANGE { TYPE | TIME | SEQUENCE } COLUMN `<column_name>`

For a target table that logs the loading history, these parameters specify the target column names that will show the change type and corresponding timestamp for each operation. The `CHANGE TYPE COLUMN <column_name>` displays I, U, or D for INSERT, UPSERT, or DELETE. In the case when multiple operations of the same type occur on the same source row with the same timestamp (because the operations are in the same transaction), use the `CHANGE SEQUENCE COLUMN <column_name>`, which adds an incremental digit to distinguish the operations.

The load behavior options are:

UPSERT: INSERT and UPDATE apply as is, DELETE converts to UPDATE

INSERT: INSERT applies as is, UPDATE and DELETE convert to INSERT

ARCHIVE: INSERT, UPDATE, and DELETE apply as is, ARCHIVE rows with a change type code of M are flagged as M in the CHANGE_TYPE column

The following example is for UPSERT for a remote subscription called `user.subscription` on a source table called `SQLServer_dbo.table`. The target table `user.table` includes a column called CHANGE_TYPE (with a data type of VARCHAR or NVARCHAR) and a column CHANGE_TIME (with a data type of TIMESTAMP).

```
CREATE REMOTE SUBSCRIPTION user.subscription
      ON "user"."SQLServer_dbo.table"
      TARGET TABLE user.table
      CHANGE TYPE COLUMN "CHANGE_TYPE"
      CHANGE TIME COLUMN "CHANGE_TIME"
      UPSERT;
```

The following example for INSERT is for the same remote subscription and includes the CHANGE_SEQUENCE column.

```
CREATE REMOTE SUBSCRIPTION user.subscription
      ON "user"."SQLServer_dbo.table"
      TARGET TABLE user.table
      CHANGE TYPE COLUMN "CHANGE_TYPE"
      CHANGE TIME COLUMN "CHANGE_TIME"
      CHANGE SEQUENCE COLUMN "CHANGE_SEQUENCE"
      INSERT;
```

<task_spec>

The task definition.

```
<task_spec>::= TARGET TASK[ <schema_name>].<task_name>
[(<var_list>)]
[PROCEDURE PARAMETERS{<param_list>)]
```

<var_list> specifies one or more start task variables.

```
<var_list> ::= <start_task_var>[{, <start_task_var>}...]
```

<start_task_var> specifies the name and value for a start task variable.

```
<start_task_var> ::= <var_name> => <var_value>
```

<var_name> is the name of variable that was defined within the task plan.

Variable values provided in this section will be used at runtime (for example, when executing the task using START TASK).

```
<var_name> ::= <identifier>
```

`<var_value>` is the value that should be used in place of the variable name specified when executing the task.

```
<var_value> ::= <string_literal>
```

`<param_list>` specifies one or more start task parameters.

```
<param_list> ::= <start_task_param> [{,
<start_task_param>}...]
```

```
<start_task_param> ::= <identifier>
```

If the task uses table types for input and/or output, then the task expects actual table, virtual table, or view names at runtime. These actual tables, virtual tables, or view names are specified as task parameters. Depending on the type of remote subscription being created, the task parameters may or may not need actual table, virtual table, or view names for specific parameters (see below for more details).

`<proc_spec>`

```
{ TARGET PROCEDURE
[<schema_name>.]<proc_name>[(<param_list>)] }
```

Description

The CREATE REMOTE SUBSCRIPTION statement creates a remote subscription in SAP HANA to capture changes specified on the entire virtual table or part of a virtual table using a subquery. The changed data can be applied to an SAP HANA target table or passed to a TASK or PROCEDURE if the changes require transformation. The owner of the remote subscription must have the following privileges:

- SELECT privilege on tables specified in the ON or AS `<subquery>` clauses
- INSERT, UPDATE, DELETE privileges on the target table
- EXECUTE privilege on the stored procedure
- START TASK privilege on the task

> ⓘ Note
>
> If you create a remote subscription using the CREATE REMOTE SUBSCRIPTION SQL statement, use **`technical user`** for the *Credentials Mode* parameter when creating a remote source.

Permissions

This statement requires the CREATE REMOTE SUBSCRIPTION object privilege on the remote source.

Remote Subscription for TARGET TASK or TARGET TABLE Using ON Clause

```
CREATE REMOTE SUBSCRIPTION [<schema_name>.]<subscription_name>
ON [<schema_name>.]<virtual_table_name>
TARGET TASK [<schema_name>].<task_name>[(<var_list>)] [PROCEDURE
PARAMETERS(<param_list>)]
```

<param_list> must contain one of the parameters as [<schema_name>.]<virtual_table_name>. This parameter must be the same schema and virtual table name as specified in the ON clause. Only one parameter in <param_list> can be a virtual table.

Each parameter in <param_list> is used in comparing its columns with columns for the corresponding table type defined in the task plan. Hence, the order of parameters in <param_list> must match the order of table types defined in the task plan for input and output sources.

The task plan table type corresponding to the procedure parameter [<schema_name>.]<virtual_table_name> must have the same columns (excluding _OP_CODE and _COMMIT_TIMESTAMP). This table type must have _OP_CODE as the last but one column and _COMMIT_TIMESTAMP as the last column.

Remote Subscription for TARGET TASK or TARGET TABLE Using AS Clause

```
CREATE REMOTE SUBSCRIPTION [<schema_name>.]<subscription_name>
AS <subquery>
TARGET TASK [<schema_name>].<task_name>[(<var_list>)] [PROCEDURE
PARAMETERS(<param_list>)]
```

The AS (<subquery>) part of the syntax lets you define the SQL and the columns to use for the subscription. The subquery should be a simple SELECT <column_list> from <virtual_table> and shouldn't contain a WHERE clause. The <column_list> should match the target table schema in column order and name.

<param_list> must contain one of the parameters as table type and this table type (schema and name) must be the same as the one defined in the task plan. This table type must also have the same columns as being output by the subquery (excluding _OP_CODE and _COMMIT_TIMESTAMP). This table type must have _OP_CODE as the last but one column and _COMMIT_TIMESTAMP as the last column. Only one parameter in <param_list> can be a table type.

Each parameter in <param_list> is used in comparing its columns with columns for the corresponding table type defined in the task plan. Hence, the order of parameters in <param_list> must match the order of table types defined in task plan for input and output sources.

Example

Create a remote subscription on a virtual table and apply changes using a real-time task.

```
CREATE SCHEMA "IM_SERVICES";
DROP REMOTE SOURCE "OracleAdapter" CASCADE;
CREATE REMOTE SOURCE "OracleAdapter" ADAPTER "OracleAdapter" AT LOCATION
dpserver CONFIGURATION '' WITH CREDENTIAL TYPE 'PASSWORD' USING '';
```

```
DROP TABLE "SYSTEM"."VT_EMPLOYEE_PK_TABLE";
CREATE VIRTUAL TABLE "SYSTEM"."VT_EMPLOYEE_PK_TABLE" AT
"OracleAdapter"."<NULL>"."<NULL>"."employee_pk_table";
DROP TYPE "IM_SERVICES"."TT_PARAM_IN";
DROP TYPE "IM_SERVICES"."TT_PARAM_OUT";
CREATE TYPE "IM_SERVICES"."TT_PARAM_IN"   AS TABLE ("empno" integer, "deptid"
integer, "empname" VARCHAR(200), "salary" decimal(28,7), "bonus" double,
"_OP_CODE" VARCHAR(1),"_COMMIT_TIMESTAMP" SECONDDATE);
CREATE TYPE "IM_SERVICES"."TT_PARAM_OUT"  AS TABLE ("empno" integer, "deptid"
integer, "empname" VARCHAR(200), "salary" decimal(28,7), "bonus" double);
DROP TABLE "IM_SERVICES"."T_OUT";
CREATE COLUMN TABLE "IM_SERVICES"."T_OUT" LIKE "IM_SERVICES"."TT_PARAM_OUT" ;
DROP TASK "IM_SERVICES"."TSKM_RT_VAR";
DROP REMOTE SUBSCRIPTION "IM_SERVICES"."RSUB_VAR";
CREATE REMOTE SUBSCRIPTION "IM_SERVICES"."RSUB_VAR"
AS (SELECT "empno","deptid","empname","salary","bonus" FROM
"SYSTEM"."VT_EMPLOYEE_PK_TABLE")
TARGET TASK "IM_SERVICES"."TSKM_RT_VAR" ("expr_var01_in1" => '100',
"expr_var02_in2" => 'upper(''walkerIN'')')
PROCEDURE PARAMETERS ( "IM_SERVICES"."TT_PARAM_IN", "IM_SERVICES"."T_OUT");
DROP REMOTE SUBSCRIPTION "IM_SERVICES"."RSUB_VAR";
CREATE REMOTE SUBSCRIPTION "IM_SERVICES"."RSUB_VAR"
ON "SYSTEM"."VT_EMPLOYEE_PK_TABLE"
TARGET TASK "IM_SERVICES"."TSKM_RT_VAR" ("expr_var01_in1" => '100',
"expr_var02_in2" => 'upper(''walkerIN'')')
PROCEDURE PARAMETERS ( "SYSTEM"."VT_EMPLOYEE_PK_TABLE", "IM_SERVICES"."T_OUT");
SELECT * FROM "SYS"."REMOTE_SUBSCRIPTIONS_";
TRUNCATE TABLE "IM_SERVICES"."T_OUT";
ALTER REMOTE SUBSCRIPTION "IM_SERVICES"."RSUB_VAR" QUEUE;
ALTER REMOTE SUBSCRIPTION "IM_SERVICES"."RSUB_VAR" DISTRIBUTE;
```

Related Information

SQL Notation Conventions (SAP HANA SQL and System Views Reference)
Data Types (SAP HANA SQL and System Views Reference)

8.2.12 CREATE VIRTUAL PROCEDURE Statement [Smart Data Integration]

Creates a virtual procedure using the specified programming language that allows execution of the procedure body at the specified remote source.

The CREATE VIRTUAL PROCEDURE SQL statement is available for use in other areas of SAP HANA, not only SAP HANA smart data integration. Refer to the CREATE VIRTUAL PROCEDURE Statement (Procedural) topic for complete information. The information in this topic is specific to smart data integration functionality.

Syntax

```
CONFIGURATION <configuration_json_string>
```

Syntax Elements

<configuration_json_string>

A JSON string that includes required source procedure parameters.

Description

The CREATE VIRTUAL PROCEDURE statement creates a new virtual procedure from a remote source procedure. When creating a virtual procedure using the SQL Console:

1. Return the metadata of the source procedure [number, types, and configuration (JSON) string] by invoking the built-in SAP HANA procedure:

   ```
   call "PUBLIC"."GET_REMOTE_SOURCE_FUNCTION_DEFINITION"
   ('<remote_source_name>','<remote_object_unique_name>',?,?,?);
   ```

2. Edit the CONFIGURATION JSON string to include the appropriate parameter values.

Permissions

This statement requires the CREATE VIRTUAL PROCEDURE object privilege on the remote source.

Example

If you use the SQL Console to create a virtual procedure, the following example illustrates an ABAP adapter.

```
CREATE VIRTUAL PROCEDURE BAPI_BANK_GETLIST (
IN BANK_CTRY NVARCHAR(6) ,
IN MAX_ROWS INT,
OUT RETURN_TYPE NVARCHAR (2),
OUT RETURN_ID NVARCHAR (40),
OUT RETURN_NUMBER VARCHAR (6) ,
OUT RETURN_MESSAGE NVARCHAR (440) ,
OUT RETURN_LOG_NO NVARCHAR (40),
OUT RETURN_LOG_MSG_NO VARCHAR (12),
OUT RETURN_MESSAGE_V1 NVARCHAR (100) ,
OUT RETURN_MESSAGE_V2 NVARCHAR (100),
OUT RETURN_MESSAGE_V3 NVARCHAR (100) ,
OUT RETURN_MESSAGE_V4 NVARCHAR (100),
OUT RETURN_PARAMETER NVARCHAR (64),
OUT RETURN_ROW INTEGER,
OUT RETURN_FIELD NVARCHAR (60),
OUT RETURN_SYSTEM NVARCHAR (20),
IN BANK_LIST_IN TABLE (
BANK_CTRY NVARCHAR (6),
BANK_KEY NVARCHAR (30),
BANK_NAME NVARCHAR (120) ,
CITY NVARCHAR (70)
),
OUT BANK_LIST TABLE (
```

```
BANK_CTRY NVARCHAR (6) ,
BANK_KEY NVARCHAR (30) ,
BANK_NAME NVARCHAR (120) ,
CITY NVARCHAR (70)
)
) CONFIGURATION '
{
"__DP_UNIQUE_NAME__": "BAPI_BANK_GETLIST",
"__DP_VIRTUAL_PROCEDURE__": true
}' AT "QA1";
```

Then call the procedure as follows:

```
CALL bapi_bank_getlist('DE', 1000, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
bank_list, ?);

[where "bank_list" is a table of type

TABLE (
                BANK_CTRY NVARCHAR (6),
                BANK_KEY NVARCHAR (30),
                BANK_NAME NVARCHAR (120) ,
                CITY NVARCHAR (70)
                )

]
```

For more information about using the SQL Console, see the *SAP HANA Administration Guide*.

8.2.13 DROP ADAPTER Statement [Smart Data Integration]

The DROP ADAPTER statement removes an adapter from all locations.

Syntax

```
DROP ADAPTER <adapter_name> [<drop_option>]
```

Syntax Elements

<adapter_name>

The name of the adapter to be dropped.

```
<adapter_name> ::= <identifier>
```

<drop_option>

When <drop_option> is not specified, a restrict drop will be performed.

```
<drop_option> ::= CASCADE | RESTRICT
```

CASCADE drops the adapter and dependent objects.

Description

The DROP ADAPTER statement removes an adapter from all locations.

Permissions

Role: sap.hana.im.dp.monitor.roles::Operations

Application privilege: sap.hana.im.dp.monitor::RemoveLocationFromAdapter

System privilege: ADAPTER ADMIN

Example

Create two agents and an adapter at both the agents.

```
CREATE AGENT TEST_AGENT_1 PROTOCOL 'TCP' HOST 'test_host1' PORT 5050;
CREATE AGENT TEST_AGENT_2 PROTOCOL 'HTTP';
CREATE ADAPTER TEST_ADAPTER AT LOCATION AGENT TEST_AGENT_1;
ALTER ADAPTER TEST_ADAPTER ADD LOCATION AGENT TEST_AGENT_2;
--Drop adapter TEST_ADAPTER.
DROP ADAPTER TEST_ADAPTER;
```

8.2.14 DROP AGENT Statement [Smart Data Integration]

The DROP AGENT statement removes an agent.

Syntax

```
DROP AGENT <agent_name> [<drop_option>]
```

Syntax Elements

<agent_name>

The name of the agent to be dropped.

```
<agent_name> ::= <identifier>
```

<drop_option>

When <drop_option> is not specified, a restrict drop is performed.

```
<drop_option> ::= CASCADE | RESTRICT
```

CASCADE drops the agent and its dependent objects.

RESTRICT drops the agent only if it does not have any dependent objects.

Description

The DROP AGENT statement removes an agent.

Permissions

Role: sap.hana.im.dp.monitor.roles::Operations

Application privilege: sap.hana.im.dp.monitor::DropAgent

System privilege: AGENT ADMIN

Example

Create an agent TEST_AGENT and adapter CUSTOM_ADAPTER on the agent. Make sure that the custom adapter is setup on the agent.

```
CREATE AGENT TEST_AGENT PROTOCOL 'TCP' HOST 'test_host' PORT 5050;
CREATE ADAPTER CUSTOM_ADAPTER AT LOCATION AGENT TEST_AGENT;
```

Drop the agent called TEST_AGENT.

```
DROP AGENT TEST_AGENT;
```

8.2.15 DROP AGENT GROUP Statement [Smart Data Integration]

The DROP AGENT GROUP statement removes an agent clustering group.

Syntax

```
DROP AGENT GROUP <agent_group_name>
```

Syntax Elements

<agent_group_name>

The name of the agent group to be dropped.

```
<agent_group_name> ::= <identifier>
```

Description

The DROP AGENT GROUP statement removes an agent clustering group. All dependent objects must be removed before an agent clustering group can be dropped.

Permissions

Role: sap.hana.im.dp.monitor.roles::Operations

Application privilege: sap.hana.im.dp.monitor::DropAgent

System privilege: AGENT ADMIN

Examples

Create an agent group TEST_GROUP.

```
CREATE AGENT GROUP TEST_GROUP;
```

Drop the agent group called TEST_GROUP.

```
DROP AGENT GROUP TEST_GROUP;
```

8.2.16 DROP REMOTE SUBSCRIPTION Statement [Smart Data Integration]

The DROP REMOTE SUBSCRIPTION statement drops an existing remote subscription.

Syntax

```
DROP REMOTE SUBSCRIPTION [<schema_name>.]<subscription_name>
```

Syntax Elements

<subscription_name>

The name of the remote subscription.

```
<subscription_name> ::= <identifier>
```

Description

The DROP REMOTE SUBSCRIPTION statement drops an existing remote subscription. If the remote subscription is actively receiving changes from source table, then a RESET command is automatically called before dropping it.

Permissions

This statement requires the DROP object privilege on the remote source.

Example

Drop the remote subscription TEST_SUB.

```
DROP REMOTE SUBSCRIPTION TEST_SUB;
```

8.2.17 GRANT Statement [Smart Data Integration]

GRANT is used to grant privileges and structured privileges to users and roles. GRANT is also used to grant roles to users and other roles.

The GRANT SQL statement is available for use in other areas of SAP HANA, not only SAP HANA smart data integration. Refer to the GRANT topic for complete information. The information below is specific to smart data integration functionality.

Syntax

Refer to the GRANT topic for complete information about GRANT syntax.

Syntax Elements

Syntax elements specific to smart data integration are described below. For information about syntax elements that are not specific to smart data integration, refer to the GRANT topic.

<system_privilege>

System privileges are used to restrict administrative tasks.

```
<system_privilege> ::= ADAPTER ADMIN | AGENT ADMIN
```

The table below describes the supported system privileges.

System Privilege	Privilege Purpose
ADAPTER ADMIN	Controls the execution of the following adapter-related commands: CREATE ADAPTER, DROP ADAPTER and ALTER ADAPTER. Also allows access to ADAPTERS and ADAPTER_LOCATIONS system views.
AGENT ADMIN	Controls the execution of the following agent-related commands: CREATE AGENT, DROP AGENT and ALTER AGENT. Also allows access to AGENTS and ADAPTER_LOCATIONS system views.

<source_privilege>

Source privileges are used to restrict the access and modifications of a source entry.

```
<source_privilege> ::= CREATE REMOTE SUBSCRIPTION | PROCESS
REMOTE SUBSCRIPTION EXCEPTION
```

Source Privilege	Privilege Purpose
CREATE REMOTE SUBSCRIPTION	This privilege allows the creation of remote subscriptions executed on this source entry. Remote subscriptions are created in a schema and point to a virtual table or SQL on tables to capture change data.
PROCESS REMOTE SUBSCRIPTION EXCEPTION	This privilege allows processing exceptions on this source entry. Exceptions that are relevant for all remote subscriptions are created for a remote source entry.

<object_privilege>

Object privileges are used to restrict the access and modifications on database objects. Database objects are tables, views, sequences, procedures, and so on.

```
<object_privilege> ::=  AGENT MESSAGING | PROCESS REMOTE
SUBSCRIPTION EXCEPTION
```

The table below describes the supported object privileges.

Object Privilege	Privilege Purpose	Command Types
AGENT MESSAGING	Authorizes the user with which the agent communicates with the data provisioning server using HTTP protocol.	DDL
PROCESS REMOTE SUBSCRIPTION EXCEPTION	Authorizes processing exceptions of a remote subscription.	DDL

Not all object privileges are applicable to all kinds of database objects. To learn which object types allow which privilege to be used, see the table below.

Privilege	Schema	Table	View	Sequence	Function / Procedure	Remote Subscription	Agent
AGENT MESSAGING	--	--	--	--	--	--	YES
PROCESS REMOTE SUBSCRIPTION EXCEPTION	--	--	--	--	--	YES	--

Related Information

GRANT Statement (Access Control) (SAP HANA SQL and System Views Reference)

8.2.18 PROCESS REMOTE SUBSCRIPTION EXCEPTION Statement [Smart Data Integration]

The PROCESS REMOTE SUBSCRIPTION EXCEPTION statement allows the user to indicate how an exception should be processed.

Syntax

```
PROCESS REMOTE SUBSCRIPTION EXCEPTION <exception_id> { RETRY | IGNORE }
```

Syntax Elements

<exception_id>

The exception ID for remote subscription or remote source.

```
<exception_id> ::= <integer_literal>
```

RETRY Indicates to retry the current failed operation. If the failure is due to opening a connection to a remote source, then the connection is established. If the failure happens when applying changed data to a target table, then the RETRY operation retries the transaction again on the target table.

IGNORE Indicates to ignore the current failure. If the failure happens when applying changed data to a target table, then the IGNORE operation skips the current transaction and proceeds with the next transaction. The exception is cleared.

Description

The PROCESS REMOTE SUBSCRIPTION EXCEPTION statement allows the user to indicate how an exception should be processed.

Permissions

This statement requires the PROCESS REMOTE SUBSCRIPTION EXCEPTION object privilege on the remote source.

Example

Ignore exception 101.

```
PROCESS REMOTE SUBSCRIPTION EXCEPTION 101 IGNORE;
```

8.2.19 START TASK Statement [Smart Data Integration]

Starts a task.

Syntax

```
START TASK <task_name> [ASYNC] [(<var_list>)] [PROCEDURE
PARAMETERS(<param_list>)]
```

Syntax Elements

<task_name>

The identifier of the task to be called, with optional schema name.

```
<task_name> ::= [<schema_name>.]<identifier>
<schema_name> ::= <identifier>
```

<var_list>

Specifies one or more start task variables. Variables passed to a task are scalar constants. Scalar parameters are assumed to be NOT NULL.

```
<var_list> ::= <start_task_var>[{, <start_task_var>}...]
```

<start_task_var>

Specifies the name and value for a start task variable. A task can contain variables that allow for dynamic replacement of task plan parameters. This section is where, at run time during START TASK, the values that should be used for those variables can be provided.

```
<start_task_var> ::= <var_name> =>
<var_value>
```

<var_name>

Name of variable that was defined.

```
<var_name> ::= <identifier>
```

<var_value>

Value that should be used in place of the variable name specified when executing the task.

```
<var_value> ::= <string_literal>
```

> **Note**
>
> **_taskhints**
>
> The reserved start task variable _taskhints can be specified in the to pass hints and settings when starting a task. The value must be a JSON string with one or more of the following properties:
>
> - qo_pop_hints
>   A comma-separated SQL optimizer hints applied to the generated SQL statement. This applies when the task is optimized into SQL (INSERT...SELECT).
>   For more information on available hint names, see HINT Details.
> - disabled_patterns
>   Calculation engine settings passed to the calculation engine. This applies when the task is executed as a calculation scenario.
> - task_hints
>   Task execution hints that control task framework behavior. This always applies.
>
> Supported task hints:
>
> `<STARTTASK_USE_CE_EXECUTION>`
>
> Forces the task to execute through the calculation engine instead of using SQL optimization. When specified, qo_pop_hints has no effect and disabled_patterns applies.
>
> **Note:**When `<STARTTASK_USE_CE_EXECUTION >`is specified, SQL optimization is bypassed, so any qo_pop_hints values are ignored.

<param_list>

Specifies one or more start task parameters.

```
<param_list> ::= <start_task_param>[{, <start_task_param>}...]
```

```
 <start_task_param> ::= <identifier>
```

Task parameters. If the task uses table types for input and/or output, then those need to be specified within this section. For more information about these data types, see BNF Lowest Terms Representations and Data Types in the Notation topic.

Parameters are implicitly defined as either IN or OUT, as inferred from the task plan. Arguments for IN parameters could be anything that satisfies the schema of the input table type (for example, a table variable internal to the procedure, or a temporary table). The actual value passed for tabular OUT parameters can be, for example, '?', a physical table name, or a table variable defined inside the procedure.

Description

Starts a task.

START TASK when executed by the client the syntax behaves in a way consistent with the SQL standard semantics, e.g. Java clients can call a procedure using a JDBC CallableStatement. Scalar output variables are a scalar value that can be retrieved from the callable statement directly.

> ⓘ Note
>
> Unquoted identifiers are implicitly treated as uppercase. Quoting identifiers will respect capitalization and allow for using white spaces which are normally not allowed in SQL identifiers.

Permissions

This statement requires the EXECUTE privilege on the schema in which the task was created.

Examples

The TASK performTranslation was already created, and the task plan has two table type input parameters and a single table type output parameter. You call the performTranslation task passing in the table types to use for execution.

```
START TASK performTranslation PROCEDURE PARAMETERS (in1, in2, out1);
```

The`< _taskhints>` variable can pass hints as part of the start task variable list:

> ✤ Example
>
> Pass SQL hints:
>
> ```
> START TASK "IM_SERVICES"."SIMPLE_TASK" ('expr_var01' => '''XYZ''',
> '_taskhints' => '{"qo_pop_hints": "IGNORE_PLAN_CACHE, USE_OLAP_PLAN"}');
> ```
>
> Force calculation engine execution:
>
> ```
> START TASK "IM_SERVICES"."SIMPLE_TASK" ('expr_var01' => '''XYZ''',
> '_taskhints' => '{"task_hints": "STARTTASK_USE_CE_EXECUTION"}');
> ```
>
> Pass SQL hints, calculation engine settings, and task hints:
>
> ```
> START TASK "IM_SERVICES"."SIMPLE_TASK" ('expr_var01'
> => '''XYZ''', '_taskhints' => '{"disabled_patterns":
> "123","qo_pop_hints": "IGNORE_PLAN_CACHE, USE_OLAP_PLAN","task_hints":
> "STARTTASK_USE_CE_EXECUTION"}');
> ```

SQL Script

You can call START TASK within the SQL Script CREATE PROCEDURE. Refer to the SAP HANA SQL Script Reference for complete details about CREATE PROCEDURE.

<proc_sql> now includes <start_task>:

```
<proc_sql> ::= <subquery>
    | <select_into_stmt>
    | <insert_stmt>
    | <delete_stmt>
     |  <update_stmt>
     |  <replace_stmt>
    | <call_stmt>
    | <create_table>
    | <drop_table>
    | <start_task>
```

START TASK is not supported in:

- Table UDF
- Scalar UDF
- Trigger
- Read-only procedures

TASK_EXECUTION_ID session variable

The TASK_EXECUTION_ID session variable provides a unique task execution ID. Knowing the proper task execution ID is critical for various pieces of task functionality including querying for side-effect information and task processing status, and canceling a task.

TASK_EXECUTION_ID is a read-only session variable. Only the internal start task code updates the value.

The value of TASK_EXECUTION_ID will be set during the START TASK command execution. In the case of asynchronous execution (START TASK ASYNC), the value is updated before the command returns so it is available before the actual task has finished asynchronously running. If the execution of START TASK was successful, then the value is updated to the unique execution ID for that START TASK execution. If the execution of START TASK was unsuccessful, then the TASK_EXECUTION_ID variable will be set back to the state as if no START TASK was run.

The users can obtain the value of TASK_EXECUTION_ID by using either of the following:

- The already existing SESSION_CONTEXT() function. If this function is used and if no tasks have been run or a task was run and it was unsuccessful, then a NULL value will be returned.
- The M_SESSION_CONTEXT monitoring view. This would need to be queried using a KEY value of “TASK_EXECUTION_ID”. If no row exists with that key, then that means that the session variable hasn’t been set (no tasks run or last task execution was unsuccessful).

> **Note**
>
> Session variables are string values. The user needs to cast appropriately based on how they want to use the value.

Table 46: Examples

Action	SQL
Obtain the last task execution ID	`SELECT SESSION_CONTEXT('TASK_EXECUTION_ID') FROM dummy;`
See monitoring information for the last task that was executed (with type casting)	`SELECT * FROM M_TASKS WHERE TASK_EXECUTION_ID = CAST (SESSION_CONTEXT('TASK_EXECUTION_ID') AS BIGINT);`
Cancel the last task that was executed (with type casting)	`CANCEL TASK CAST(SESSION_CONTEXT('TASK_EXECUTION_ID') AS BIGINT);`

Related Information

SQL Notation Conventions (SAP HANA SQL and System Views Reference)

8.3 System Views

System views allow you to query for various information about the system state using SQL commands. The results appear as tables.

System views are located in the SYS schema. In a system with tenant databases, every database has a SYS schema with system views that contain information about that database only. In addition, the system database has a further schema, SYS_DATABASES, which contains views for monitoring the system as a whole. The views in the SYS_DATABASES schema provide aggregated information from a subset of the views available in the SYS schema of all tenant databases in the system. These union views have the additional column DATABASE_NAME to allow you to identify to which database the information refers. To be able to view information in these views, you need the system privilege CATALOG READ or DATABASE ADMIN.

SAP HANA system views are separated into two categories: **metadata** views and **runtime** views. Metadata views provide metadata about objects in the database, including options or settings that were set using a DDL statement. Runtime views provide actual HANA runtime data, including statistics and status information related to the execution of DML statements. Runtime views start with M_ for monitoring.

ADAPTER_CAPABILITIES System View [Smart Data Integration] [page 305]
: Specifies the SQL capabilities of the adapters stored in the system.

ADAPTER_LOCATIONS System View [Smart Data Integration] [page 305]
: Specifies the location of adapters.

ADAPTERS System View [Smart Data Integration] [page 306]
: Stores adapters available in the SAP HANA system.

AGENT_CONFIGURATION System View [Smart Data Integration] [page 306]
: Agent configuration

AGENT_GROUPS System View [Smart Data Integration] [page 307]
: Lists active data provisioning agent groups in the system.

AGENTS System View [Smart Data Integration] [page 307]
Lists active data provisioning agents in the system.

M_AGENTS System View [Smart Data Integration] [page 308]
Provides the status of all agents registered in the SAP HANA database.

M_REMOTE_SOURCES System View [Smart Data Integration] [page 308]
Stores dictionary status information, remote source owner information, and the status of data collection.

M_REMOTE_SUBSCRIPTION_COMPONENTS System View [Smart Data Integration] [page 309]
Provides the status of a remote subscription for each internal component.

M_REMOTE_SUBSCRIPTION_STATISTICS System View [Smart Data Integration] [page 310]
Provides details of current processing details of a remote subscription (e.g. number of messages or transactions received, applied since the start of the SAP HANA database).

M_REMOTE_SUBSCRIPTIONS System View [Smart Data Integration] [page 310]
Provides the status and run-time information of a remote subscription.

M_REMOTE_QUERY_STATISTICS System View [Smart Data Integration] [page 312]
Provides monitoring statistics for remote source queries.

M_SESSION_CONTEXT System View [Smart Data Integration] [page 314]
Session variables for each connection

REMOTE_SOURCE_OBJECT_COLUMNS System View [Smart Data Integration] [page 315]
If the adapter can provide column-level information for each table in the remote system, once this dictionary is built you can search for relationships between tables. This table is useful for analyzing relationships between tables in the remote source.

REMOTE_SOURCE_ OBJECT_DESCRIPTIONS System View [Smart Data Integration] [page 316]
Stores description of browsable node in different languages.

REMOTE_SOURCE_OBJECTS System View [Smart Data Integration] [page 316]
Stores browsable nodes as well as importable objects (virtual tables). This view is built from remote source metadata dictionaries.

REMOTE_SOURCES System View [Smart Data Integration] [page 317]
Remote sources

REMOTE_SUBSCRIPTION_EXCEPTIONS System View [Smart Data Integration] [page 317]
Provides details about an exception that occurred during the execution of a remote subscription. The exceptions can be processed using the PROCESS REMOTE SUBSCRIPTION EXCEPTION SQL statement.

REMOTE_SUBSCRIPTIONS System View [Smart Data Integration] [page 318]
Lists all the remote subscriptions created for a remote source.

TASK_CLIENT_MAPPING System View [Smart Data Integration] [page 319]
Provides the client mapping when a task is created by the ABAP API.

TASK_COLUMN_DEFINITIONS System View [Smart Data Integration] [page 319]
Defines the columns present in a particular table.

TASK_EXECUTIONS System View [Smart Data Integration] [page 320]
Task-level run-time statistics generated when START TASK is run.

TASK_LOCALIZATION System View [Smart Data Integration] [page 321]
Contains localized values for the task framework tables.

TASK_OPERATIONS System View [Smart Data Integration] [page 321]
Contains all operations that exist for a given task, as well as details about those operations.

TASK_OPERATIONS_EXECUTIONS System View [Smart Data Integration] [page 322]
Operations-level task statistics generated when START TASK is run.

TASK_PARAMETERS System View [Smart Data Integration] [page 323]
Details about the task parameters view

TASK_TABLE_DEFINITIONS System View [Smart Data Integration] [page 323]
Contains all of the tables used by the various side-effect producing operation.

TASK_TABLE_RELATIONSHIPS System View [Smart Data Integration] [page 324]
Defines the relationships, if any, between tables within an operation.

TASKS System View [Smart Data Integration] [page 325]
Details about tasks.

VIRTUAL_COLUMN_PROPERTIES System View [Smart Data Integration] [page 326]
Lists the properties of the columns in a virtual table sent by the adapter via CREATE VIRTUAL TABLE SQL statement.

VIRTUAL_TABLE_PROPERTIES System View [Smart Data Integration] [page 326]
Lists the properties of a virtual table sent by the adapter via the CREATE VIRTUAL TABLE SQL statement.

BEST_RECORD_GROUP_MASTER_STATISTICS System View [Smart Data Quality] [page 327]
Contains a summary of Best Record group master statistics.

BEST_RECORD_RESULTS System View [Smart Data Quality] [page 328]
Contains governance information for every column in every record that is updated in the best record process.

BEST_RECORD_STRATEGIES System View [Smart Data Quality] [page 329]
Contains information on which strategies are used in each strategy group and in which order.

CLEANSE_ADDRESS_RECORD_INFO System View [Smart Data Quality] [page 329]
Describes how well an address was assigned as well as the type of address.

CLEANSE_CHANGE_INFO System View [Smart Data Quality] [page 330]
Describes the changes made during the cleansing process.

CLEANSE_COMPONENT_INFO System View [Smart Data Quality] [page 331]
Identifies the location of parsed data elements in the input and output.

CLEANSE_INFO_CODES System View [Smart Data Quality] [page 333]
Contains one row per info code generated by the cleansing process.

CLEANSE_STATISTICS System View [Smart Data Quality] [page 334]
Contains a summary of Cleanse statistics.

GEOCODE_INFO_CODES System View [Smart Data Quality] [page 334]
Contains one row per info code generated by the geocode transformation process.

GEOCODE_STATISTICS System View [Smart Data Quality] [page 335]
Contains a summary of Geocode statistics.

MATCH_GROUP_INFO System View [Smart Data Quality] [page 336]
Contains one row for each match group.

MATCH_RECORD_INFO System View [Smart Data Quality] [page 336]

Contains one row for each matching record per level.

MATCH_SOURCE_STATISTICS System View [Smart Data Quality] [page 337]
Contains counts of matches within and between data sources.

MATCH_STATISTICS System View [Smart Data Quality] [page 338]
Contains statistics regarding the run of the transformation operation.

MATCH_TRACING System View [Smart Data Quality] [page 339]
Contains one row for each match decision made during the matching process.

Related Information

System Views Reference for Additional SAP HANA Contexts (SAP HANA SQL and System Views Reference)

8.3.1 ADAPTER_CAPABILITIES System View [Smart Data Integration]

Specifies the SQL capabilities of the adapters stored in the system.

Structure

Column	Data type	Description
ADAPTER_NAME	NVARCHAR(64)	Adapter name
SOURCE_VERSION	NVARCHAR(64)	Source versions supported by the adapter

8.3.2 ADAPTER_LOCATIONS System View [Smart Data Integration]

Specifies the location of adapters.

Structure

Column	Data type	Description
ADAPTER_NAME	NVARCHAR(64)	Adapter name

Column	Data type	Description
LOCATION	VARCHAR(11)	Location of the adapter: 'indexserver', 'dpserver', 'agent'
AGENT_NAME	NVARCHAR(256)	Agent name

8.3.3 ADAPTERS System View [Smart Data Integration]

Stores adapters available in the SAP HANA system.

Structure

Column	Data type	Description
ADAPTER_NAME	NVARCHAR(64)	Adapter name
PROPERTIES	NVARCHAR(1000)	Optional properties of the adapter such as display_name and description
CONFIGURATION	NCLOB	UI properties that must be displayed when configuring remote data source
IS_SYSTEM_ADAPTER	VARCHAR(5)	Specifies whether the adapter is a system adapter: 'TRUE'/'FALSE'
IS_ESS_DEFINITION_SUPPORTED	VARCHAR(5)	Specifies if the procedure GET_REMOTE_SOURCE_TABLE_ESS_DEFINITIONS is enabled for remote sources created using this adapter: 'TRUE'/'FALSE'

8.3.4 AGENT_CONFIGURATION System View [Smart Data Integration]

Agent configuration

Structure

Column name	Data type	Description
AGENT_NAME	NVARCHAR(256)	Agent name
KEY	VARCHAR(128)	Agent property key

Column name	Data type	Description
VALUE	NCLOB	Agent property value

8.3.5 AGENT_GROUPS System View [Smart Data Integration]

Lists active data provisioning agent groups in the system.

Structure

Column	Data type	Description
AGENT_GROUP_NAME	NVARCHAR(256)	Name of the agent group.

8.3.6 AGENTS System View [Smart Data Integration]

Lists active data provisioning agents in the system.

Structure

Column	Data type	Description
AGENT_NAME	NVARCHAR(256)	Agent name
PROTOCOL	VARCHAR(4)	Protocol for communication with SAP HANA database: 'TCP', 'HTTP'
AGENT_HOST	NVARCHAR (64)	Agent host specified when using TCP
AGENT_PORT	INTEGER	Agent port specified when using TCP
IS_SSL_ENABLED	VARCHAR(5)	Specifies whether the agent listening on TCP port uses SSL
AGENT_GROUP_NAME	NVARCHAR(256)	Agent clustering group to which the agent belongs.

8.3.7 M_AGENTS System View [Smart Data Integration]

Provides the status of all agents registered in the SAP HANA database.

Structure

Column	Data type	Description
AGENT_NAME	NVARCHAR(256)	Agent name
FREE_PHYSICAL_MEMORY	BIGINT	Free physical memory on the host
FREE_SWAP_SPACE	BIGINT	Free swap memory on the host
LAST_CONNECT_TIME	TIMESTAMP	The last time the session cookie was used for successful re-connection
SYS_TIMESTAMP	TIMESTAMP	Host timestamp in local time zone
USED_PHYSICAL_MEMORY	BIGINT	Used physical memory on the host
USED_SWAP_SPACE	BIGINT	Used swap memory on the host
UTC_TIMESTAMP	TIMESTAMP	Host timestamp in UTC
AGENT_VERSION	VARCHAR(32)	Agent version
AGENT_STATUS	VARCHAR(16)	Agent status

8.3.8 M_REMOTE_SOURCES System View [Smart Data Integration]

Stores dictionary status information, remote source owner information, and the status of data collection.

> ⓘ Note
>
> This system view is for keeping track of the status of metadata dictionaries for remote sources. If there is no dictionary for a given remote source, it will not appear in the view.

For basic remote source information you can select from REMOTE_SOURCES. It includes the following.

- REMOTE_SOURCE_NAME
- ADAPTER_NAME
- CONNECTION_INFO
- AGENT_GROUP_NAME

Structure

Column	Data type	Description
USER_NAME	NVARCHAR(256)	User name
REMOTE_SOURCE_NAME	NVARCHAR(256)	Remote source name
LAST_REFRESH_TIME	TIMESTAMP	The successful completion timestamp of the refresh operation
REFRESH_START_TIME	TIMESTAMP	The timestamp of when the refresh operation was executed
REFRESH_STATUS	VARCHAR(32)	Refresh operation status: • STARTED • COMPLETED • RUNNING (GET OBJECTS) • RUNNING (GET OBJECT DETAILS) • FAILED • CANCELLED • CLEARED
REFRESH_ERROR_MESSAGE	NVARCHAR(2000)	Exception message that occurred during refresh operation

8.3.9 M_REMOTE_SUBSCRIPTION_COMPONENTS System View [Smart Data Integration]

Provides the status of a remote subscription for each internal component.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Remote subscription schema name
SUBSCRIPTION_NAME	NVARCHAR(256)	Remote subscription name
COMPONENT	VARCHAR(10)	• DPSERVER • ADAPTER • RECEIVER • APPLIER
STATUS	VARCHAR	Component status
MESSAGE	VARCHAR	Additional information

8.3.10 M_REMOTE_SUBSCRIPTION_STATISTICS System View [Smart Data Integration]

Provides details of current processing details of a remote subscription (e.g. number of messages or transactions received, applied since the start of the SAP HANA database).

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Remote subscription schema name
SUBSCRIPTION_NAME	NVARCHAR(256)	Remote subscription name
RECEIVED_MESSAGE_COUNT	BIGINT	Total message/transaction count received by the current connection
RECEIVED_MESSAGE_SIZE	BIGINT	Total size of messages/transactions received by the current connection
APPLIED_MESSAGE_COUNT	BIGINT	Total number of messages/transactions applied
APPLIED_MESSAGE_SIZE	BIGINT	Total size of messages/records applied
REJECTED_MESSAGE_COUNT	BIGINT	Total number of messages/records rejected
LAST_MESSAGE_RECEIVED	TIMESTAMP	Time at which the last message/transaction is received
LAST_MESSAGE_APPLIED	TIMESTAMP	Time at which the last message/transaction is applied
RECEIVER_LATENCY	BIGINT	Receiver latency in microseconds
APPLIER_LATENCY	BIGINT	Applier latency in microseconds

8.3.11 M_REMOTE_SUBSCRIPTIONS System View [Smart Data Integration]

Provides the status and run-time information of a remote subscription.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Remote subscription schema name
SUBSCRIPTION_NAME	NVARCHAR(256)	Remote subscription name
STATE	VARCHAR(256)	State of event

Column	Data type	Description
OPTIMIZED_QUERY_STRING	NCLOB	This is generated and saved so that if there are multiple subscriptions interested in same query result, and the same internal_distribution_id, both the subscriptions can use the same result.
OPTIMIZED_QUERY_HASH	VARCHAR(128)	Hash is used to query the match for optimized query string
INTERNAL_DISTRIBUTION_ID	BIGINT	Generated integer to identify if multiple target tables are interested in the changes from same source SQL or virtual table
OPTIMIZED_QUERY_RESULTSET_TYPE	TINYINT	0 - REGULAR 1 - CLUSTER 2 - POOL
REMOTE_SUBSCRIPTION	NVARCHAR(256)	An optional subscription name registered by the adapter in the remote source system
VOLUME_ID	INTEGER	Persistence Volume ID
BEGIN_MARKER	VARCHAR(64)	Generated begin marker in the format `B<remote_source_oid>_<remote_subscription_oid>_<YYYYMMDDHH24MMSSFF7>` when QUEUE command is called.
END_MARKER	VARCHAR(64)	Generated end marker in the format `E<remote_source_oid>_<remote_subscription_oid>_<YYYYMMDDHH24MMSSFF7>` when DISTRIBUTE command is called.
BEGIN_MARKER_TIME	TIMESTAMP	Timestamp when QUEUE request is received.
END_MARKER_TIME	TIMESTAMP	Timestamp when DISTRIBUTE command is called.
LAST_PROCESSED_TRANSACTION_ID	VARBINARY(128)	Transaction ID of the last processed transaction.
LAST_PROCESSED_TRANSACTION_TIME	TIMESTAMP	Time when the last transaction was applied.
LAST_PROCESSED_BEGIN_SEQUENCE_ID	VARBINARY(68)	Last processed transaction's begin record sequence ID
LAST_PROCESSED_COMMIT_SEQUENCE_ID	VARBINARY(68)	Last processed transaction's commit record sequence ID
LAST_RECEIVED_SEQUENCE_ID	VARBINARY(68)	Last received sequence ID
LAST_RECEIVED_CUSTOM_ID	NVARCHAR(64)	Last received custom ID. Custom IDs may be used by adapters with every changed-data row of a transaction.

Column	Data type	Description
LAST_PROCESSED_CUSTOM_ID	NVARCHAR(64)	Last processed custom ID. Custom IDs may be used by adapters with every changed-data row of a transaction.

8.3.12 M_REMOTE_QUERY_STATISTICS System View [Smart Data Integration]

Provides monitoring statistics for remote source queries.

Structure

Column	Data type	Description
CONNECTION_ID	INTEGER	Connection ID
TRANSACTION_ID	INTEGER	Transaction ID
STATEMENT_ID	NVARCHAR(12)	Statement ID
REMOTE_SOURCE_SCHEMA_NAME	NVARCHAR(256)	Remote source schema name
REMOTE_SOURCE_NAME	NVARCHAR(256)	Remote source name
USER_NAME	NVARCHAR(256)	User name
ACCESSED_VIRTUAL_TABLE_NAMES	NCLOB	Relevant virtual table name list accessed by the query. The format is `<schema>.<name>` for a list entry, with entries separated by a comma.
STATEMENT_STRING	NCLOB	SAP HANA statement string
STATEMENT_HASH	NVARCHAR(32)	MD5 has value for the SAP HANA statement string
REMOTE_STATEMENT_STRING	NCLOB	Rewritten remote statement string
REMOTE_STATEMENT_STATUS	NVARCHAR(10)	Remote statement status: • STARTING • EXECUTING • FETCHING • CANCELLING • CANCELLED • ERROR • CLOSED
REMOTE_STATEMENT_DETAILS	NCLOB	Remote statement details
FETCH_SIZE	INTEGER	Number of records to fetch at a time

Column	Data type	Description
MAX_RECORD_SIZE	INTEGER	Maximum potential size of a record, in bytes
IS_STREAMING	NVARCHAR(5)	Flag indicating whether results are streamed
START_TIME	TIMESTAMP	Statement start time
END_TIME	TIMESTAMP	Time when the statement is closed
STATEMENT_REWRITE_DURATION	INTEGER	Time taken to rewrite the statement, in milliseconds
STATEMENT_EXECUTE_DURATION	INTEGER	Time taken to execute the statement, in milliseconds
RESULTSET_FETCH_DURATION	INTEGER	Time taken to fetch the result set, in milliseconds
RESULTSET_SERIALIZE_DURATION	INTEGER	Time taken to serialize the result set, in milliseconds
RESULTSET_SEND_DURATION	INTEGER	TIme taken to send the result set, in milliseconds
RECEIVED_RECORD_COUNT	BIGINT	Number of records received by the Data Provisioning Server
RECEIVED_SIZE	BIGINT	Size of the records received by the Data Provisioning Server, in bytes
DELIVERED_RECORD_COUNT	BIGINT	Number of records delivered to the Index Server
DELIVERED_RECORD_SIZE	BIGINT	Size of the records delivered to the Index Server, in bytes
AVG_REQUEST_IDLE_DURATION	INTEGER	Average time waiting for fetch requests from the Index Server, in milliseconds
AVG_REQUEST_SERVICING_DURATION	INTEGER	Average time servicing fetch requests from the Index Server, in milliseconds
AGENT_NAME	NVARCHAR(256)	Name of the agent used to process the remote query
AGENT_START_TIME	TIMESTAMP	Agent statement start time
AGENT_END_TIME	TIMESTAMP	Time when the statement is closed on the agent
AGENT_SENDING_RECORD_COUNT	BIGINT	Number of records sent by the agent
AGENT_SERIALIZER_QUEUE_PEAK_MEMORY_IN_USE	BIGINT	Peak memory used by the serializer queue, in bytes
AGENT_SERIALIZER_QUEUE_SIZE_ON_PEAK_MEMORY_USE	BIGINT	Number of objects in the serializer queue at the time of peak memory usage
AGENT_SERIALIZER_QUEUE_PEAK_TIME	TIMESTAMP	Time when the serializer queue reached its peak usage
AGENT_SERIALIZER_QUEUE_TOTAL_OBJECT_SIZE	BIGINT	Total size of all objects that passed through the serializer queue, in bytes

Column	Data type	Description
AGENT_SERIALIZER_QUEUE_TOTAL_OBJECT_COUNT	BIGINT	Total count of all objects that passed through the serializer queue
AGENT_SERIALIZER_POOL_CAPACITY	BIGINT	Capacity of the serializer shared memory pool, in bytes
AGENT_SERIALIZER_POOL_PEAK_MEMORY_IN_USE_DURING_EXECUTION	BIGINT	Peak memory used by the serializer shared memory pool during statement execution, in bytes
AGENT_SERIALIZER_POOL_PEAK_TIME	TIMESTAMP	Time when the serializer pool reached its peak size
AGENT_RESPONSE_QUEUE_PEAK_MEMORY_IN_USE	BIGINT	Peak memory used by the response queue, in bytes
AGENT_RESPONSE_QUEUE_SIZE_ON_PEAK_MEMORY_USE	BIGINT	Number of objects in the response queue at the time of peak usage
AGENT_RESPONSE_QUEUE_PEAK_TIME	TIMESTAMP	Time when the response queue reached its peak usage
AGENT_RESPONSE_QUEUE_TOTAL_OBJECT_SIZE	BIGINT	Total size of all objects that passed through the response queue, in bytes
AGENT_RESPONSE_QUEUE_TOTAL_OBJECT_COUNT	BIGINT	Total count of all objects that passed through the response queue
AGENT_RESPONSE_POOL_CAPACITY	BIGINT	Capacity of the response shared memory pool, in bytes
AGENT_RESPONSE_POOL_PEAK_MEMORY_IN_USE_DURING_EXECUTION	BIGINT	Peak memory used by the response shared memory pool during statement execution, in bytes
AGENT_RESPONSE_POOL_PEAK_TIME	TIMESTAMP	Time when the response pool reached its peak usage
PREFETCH_QUEUE_AVG_MEMORY_IN_USE	BIGINT	Average memory used by the Data Provisioning Server prefetch queue, in bytes
PREFETCH_QUEUE_PEAK_MEMORY_IN_USE	BIGINT	Peak memory used by the Data Provisioning Server prefetch queue, in bytes
PREFETCH_QUEUE_PEAK_TIME	TIMESTAMP	Time when the prefetch queue reached its peak usage

8.3.13 M_SESSION_CONTEXT System View [Smart Data Integration]

Session variables for each connection

> **Note**
>
> The M_SESSION_CONTEXT view is available for use in other areas of SAP HANA, not only SAP HANA smart data integration. Refer to the M_SESSION_CONTEXT topic for complete information. The information below is specific to smart data integration functionality.

This view shows session variables of all open connections.

Each variable is categorized in SECTION column to USER (user defined variable using SET command or client API call) or SYSTEM (predefined variable or system property).

Table 47: Predefined variables

Variable Name (M_SESSION_CONTEXT.KEY)	Value Constraint	Set by Client or Server	Shown in M_SESSION_CONTEXT	Server Usage	Description
TASK_EXECUTION_ID	bigint	server	yes	START TASK	Shows unique task execution ID

Related Information

M_SESSION_CONTEXT System View (SAP HANA SQL and System Views Reference)

8.3.14 REMOTE_SOURCE_OBJECT_COLUMNS System View [Smart Data Integration]

If the adapter can provide column-level information for each table in the remote system, once this dictionary is built you can search for relationships between tables. This table is useful for analyzing relationships between tables in the remote source.

Structure

Column	Data type	Description
USER_NAME	NVARCHAR(256)	For secondary credentials, need to know the owner name
REMOTE_SOURCE_NAME	NVARCHAR(256)	To uniquely identify a remote source
OBJECT_NAME	NVARCHAR(5000)	Unique name to identify remote source object
COLUMN_NAME	NVARCHAR(256)	Column name
DATA_TYPE_NAME	VARCHAR(16)	SAP HANA data type
REMOTE_DATA_TYPE_NAME	VARCHAR(32)	Remote source data type
REMOTE_CONTENT_TYPE	NVARCHAR(256)	Examples include address, unit of measure, user-defined types, ZIP code, and so on
LENGTH	INTEGER	Length/precision of the column
SCALE	INTEGER	Scale of the column

Column	Data type	Description
IS_NULLABLE IS_AUTOINCREMENT	VARCHAR(5)	Various column properties

8.3.15 REMOTE_SOURCE_ OBJECT_DESCRIPTIONS System View [Smart Data Integration]

Stores description of browsable node in different languages.

Structure

Column	Data type	Description
USER_NAME	NVARCHAR(256)	User name
REMOTE_SOURCE_NAME	NVARCHAR(256)	Remote source name
OBJECT_NAME	NVARCHAR(5000)	Unique name to identify remote source object
LANGUAGE_CODE	VARCHAR(2)	Language code
DESCRIPTION	NVARCHAR(5000)	Description of this object

8.3.16 REMOTE_SOURCE_OBJECTS System View [Smart Data Integration]

Stores browsable nodes as well as importable objects (virtual tables). This view is built from remote source metadata dictionaries.

Structure

Column	Data type	Description
USER_NAME	NVARCHAR(256)	User name
REMOTE_SOURCE_NAME	NVARCHAR(256)	Remote source name
OBJECT_NAME	NVARCHAR(5000)	Unique name to identify remote source object
DISPLAY_NAME	NVARCHAR(256)	Display name for this object

Column	Data type	Description
IS_IMPORTABLE	VARCHAR(5)	If the object is importable as a virtual table: 'TRUE'/'FALSE'
IS_EXPANDABLE	VARCHAR(5)	If the object can be expanded or browsed to get inner objects: 'TRUE/'FALSE'
PARENT_OBJECT_NAME	NVARCHAR(5000)	The parent object name for this object
DEFINITION_TYPE	VARCHAR(32)	Object definition type
DEFINITION	NCLOB	Object definition

8.3.17 REMOTE_SOURCES System View [Smart Data Integration]

Remote sources

Structure

Column name	Data type	Description
REMOTE_SOURCE_NAME	NVARCHAR(256)	Remote source name
ADAPTER_NAME	NVARCHAR(256)	Adapter name
CONNECTION_INFO	NVARCHAR(256)	Connection information
AGENT_GROUP_NAME	NVARCHAR(256)	Name of the agent group name

Related Information

REMOTE_SOURCES System View (SAP HANA SQL and System Views Reference)

8.3.18 REMOTE_SUBSCRIPTION_EXCEPTIONS System View [Smart Data Integration]

Provides details about an exception that occurred during the execution of a remote subscription. The exceptions can be processed using the PROCESS REMOTE SUBSCRIPTION EXCEPTION SQL statement.

Structure

Column	Data type	Description
EXCEPTION_OID	BIGINT	Exception ID
OBJECT_TYPE	VARCHAR(19)	'REMOTE SOURCE', 'REMOTE SUBSCRIPTION'
OBJECT_SCHEMA_NAME	NVARCHAR(256)	Schema name of remote source or remote subscription based on OBJECT_TYPE
OBJECT_NAME	NVARCHAR(256)	Object name of remote source or remote subscription based on OBJECT_TYPE
EXCEPTION_TIME	TIMESTAMP	Time at which the exception was raised
ERROR_NUMBER	INTEGER	Error number
ERROR_MESSAGE	NVARCHAR(2000)	Error message
COMPONENT	VARCHAR(8)	Component that raised the exception

8.3.19 REMOTE_SUBSCRIPTIONS System View [Smart Data Integration]

Lists all the remote subscriptions created for a remote source.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Remote subscription schema name
SUBSCRIPTION_NAME	NVARCHAR(256)	Remote subscription name
OWNER_NAME	NVARCHAR(256)	Owner name
REMOTE_SOURCE_NAME	NVARCHAR(256)	Remote source name
IS_VALID	VARCHAR(5)	Specifies whether the remote subscription is valid or not. This becomes FALSE when its source or target objects are changed or dropped.
SUBSCRIPTION_TYPE	VARCHAR(13)	Remote subscription type
VIRTUAL_TABLE_SCHEMA_NAME	NVARCHAR(256)	Virtual table schema name
VIRTUAL_TABLE_NAME	NVARCHAR(256)	Virtual table name
SUBSCRIPTION_QUERY_STRING	NCLOB	Select statement specified in the subscription when subscription type is SQL
TARGET_OBJECT_TYPE	VARCHAR(9)	Remote subscription target object type: 'TABLE', 'PROCEDURE', 'TASK'

Column	Data type	Description
TARGET_OBJECT_SCHEMA_NAME	NVARCHAR(256)	Target object schema name
TARGET_OBJECT_NAME	NVARCHAR(256)	Target object name
TARGET_OTHER_PARAM_STRING	NVARCHAR(4000)	Constant parameter string to pass at execution when target object type is PROCEDURE or TASK
TASK_PROCEDURE_PARAMETERS	NVARCHAR(5000)	A comma-separated list of task parameters.

8.3.20 TASK_CLIENT_MAPPING System View [Smart Data Integration]

Provides the client mapping when a task is created by the ABAP API.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
CLIENT	NVARCHAR(128)	Name of the client that created the task with the ABAP API

8.3.21 TASK_COLUMN_DEFINITIONS System View [Smart Data Integration]

Defines the columns present in a particular table.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task

Column	Data type	Description
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
TABLE_NAME	NVARCHAR(128)	Name of the table defined in the task plan for the operation
COLUMN_NAME	NVARCHAR(128)	Name of the column used in the task plan within a table
MAPPED_NAME	NVARCHAR(128)	Mapped name of the column used in a task plan within a table

8.3.22 TASK_EXECUTIONS System View [Smart Data Integration]

Task-level run-time statistics generated when START TASK is run.

TASK_EXECUTIONS shows one record per task plan execution.

Data in this view is updated while the task is in progress. For example, STATUS, PROCESSED_RECORDS, and TOTAL_PROGRESS_PERCENT are continuously updated until the task is complete.

Users may view information only for tasks that they ran themselves or were granted permissions to view.

Structure

Column	Data type	Description
HOST	VARCHAR(64)	Host name
PORT	INTEGER	Internal port
SCHEMA_NAME	NVARCHAR(256)	Schema name used in the task
TASK_NAME	NVARCHAR(256)	Name of the task
CONNECTION_ID	INTEGER	Connection identifier
TRANSACTION_ID	INTEGER	Transaction identifier used for the task execution
TASK_EXECUTION_ID	BIGINT	Task execution unique identifier
PARENT_TASK_EXECUTION_ID	BIGINT	Parent task identifier
IS_ASYNC	VARCHAR(5)	TRUE if the task is asynchronous, else FALSE
PARAMETERS	NVARCHAR(5000)	Input parameters for the task
PROCEDURE_PARAMETERS	NVARCHAR(5000)	Displays the input <param-list> values that were specified in the START TASK SQL command
START_TIME	TIMESTAMP	Start time of the task
END_TIME	TIMESTAMP	End time of the task

Column	Data type	Description
DURATION	BIGINT	Execution time of the task (microseconds)
STATUS	VARCHAR(16)	Status of the task: STARTING, RUNNING, FAILED, COMPLETED, CANCELLING, or CANCELLED
CURRENT_OPERATION	NVARCHAR(128)	Current operation of the task
PROCESSED_RECORDS	BIGINT	Total number of records processed
TOTAL_PROGRESS_PERCENT	BIGINT	Total task progress (percent)
USER_NAME	NVARCHAR(256)	User name
APPLICATION_USER_NAME	NVARCHAR(256)	Application user name
HAS SIDE EFFECTS	VARCHAR(5)	'TRUE' if the task produces side effect data, else 'FALSE'

8.3.23 TASK_LOCALIZATION System View [Smart Data Integration]

Contains localized values for the task framework tables.

Structure

Column	Data type	Description
LOC_TYPE_ID	INTEGER	Identifier of the type of the entity being localized
LOC_ID	NVARCHAR(64)	Identifier of the entity being localized
LANGUAGE	NVARCHAR(1)	One-character code of the localized language
DESCRIPTION	NVARCHAR(1024)	Localized description

8.3.24 TASK_OPERATIONS System View [Smart Data Integration]

Contains all operations that exist for a given task, as well as details about those operations.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
COMMENTS	NVARCHAR(512)	Comments made on the operation
HAS_SIDE_EFFECTS	TINYINT	Specifies whether the operation has side-effect data
OPERATION_TYPE	NVARCHAR(128)	Type of operation in the task plan

8.3.25 TASK_OPERATIONS_EXECUTIONS System View [Smart Data Integration]

Operations-level task statistics generated when START TASK is run.

TASK_OPERATIONS_EXECUTIONS shows one record per operation.

Data in this view is updated while the task is in progress. For example, STATUS, PROCESSED_RECORDS, and OPERATIONS_PROGRESS_PERCENT are continuously updated until the task is complete.

Users may view information only for tasks that they ran themselves or were granted permissions to view.

Structure

Column	Data type	Description
HOST	VARCHAR(64)	Host name
PORT	INTEGER	Internal port
TASK_EXECUTION_ID	BIGINT	Task identifier
CONNECTION_ID	INTEGER	Connection identifier
TRANSACTION_ID	INTEGER	Transaction identifier used for the task execution
CURRENT_OPERATION	NVARCHAR	Name of operation
OPERATION_TYPE	NVARCHAR(128)	Type of operation
OPERATION_NAME	NVARCHAR(128)	Internal name of operation
START_TIME	TIMESTAMP	Start time of the task
END_TIME	TIMESTAMP	End time of the task

Column	Data type	Description
DURATION	BIGINT	Execution time of the task (microseconds)
STATUS	VARCHAR(16)	Status of the task: • STARTING • RUNNING • FAILED • COMPLETED • CANCELLING • CANCELLED
PROCESSED_RECORDS	BIGINT	Total number of records processed
OPERATION_PROGRESS_PERCENT	DOUBLE	Operation progress (percent)
HAS_SIDE_EFFECTS	VARCHAR(5)	'TRUE' if the task produces side effect data, else 'FALSE'

8.3.26 TASK_PARAMETERS System View [Smart Data Integration]

Details about the task parameters view

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Schema in which the task was created
TASK_NAME	NVARCHAR(256)	Name of task
PARAMETER_NAME	NVARCHAR(256)	Name of parameter
POSITION	INTEGER	Position of parameter
TABLE_TYPE_SCHEMA	NVARCHAR(256)	Schema in which the TableType was created
TABLE_TYPE_NAME	NVARCHAR(256)	Name of TableType
PARAMETER_TYPE	VARCHAR(7)	Parameter type: IN or OUT

8.3.27 TASK_TABLE_DEFINITIONS System View [Smart Data Integration]

Contains all of the tables used by the various side-effect producing operation.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
TABLE_ID	INTEGER	Unique identifier for the table
TABLE_NAME	NVARCHAR(128)	Name of the table defined in the task plan for an operation
SIDE_EFFECT_SCHEMA	NVARCHAR(128)	Schema where the generated side-effect table is located
SIDE_EFFECT_NAME	NVARCHAR(128)	Name of the generated side-effect table
IS_PRIMARY_TABLE	TINYINT	Specifies whether this table is the primary table in a relationship
OPERATION_TABLE_TYPE	NVARCHAR(20)	Type of operation that the table is used within

8.3.28 TASK_TABLE_RELATIONSHIPS System View [Smart Data Integration]

Defines the relationships, if any, between tables within an operation.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
TABLE_NAME	NVARCHAR(128)	Name of the table defined in the task plan for an operation
RELATED_TABLE_NAME	NVARCHAR(128)	Name of the table to which the table specified in TABLE_NAME is related

Column	Data type	Description
FROM_ATTRIBUTE	NVARCHAR(128)	Name of the column in the TABLE_NAME table that relates to the TO_ATTRIBUTE
TO_ATTRIBUTE	NVARCHAR(128)	Name of the column in the RELATED_TABLE_NAME table that relates to the FROM_ATTRIBUTE

8.3.29 TASKS System View [Smart Data Integration]

Details about tasks.

Structure

Column	Data type	Description
TASK_OID	BIGINT	Unique identifier for a task
TASK_NAME	NVARCHAR(256)	Name of task
SCHEMA_NAME	NVARCHAR(256)	Schema the task was created in
OWNER_NAME	NVARCHAR(256)	Owner of the task
CREATE_TIME	TIMESTAMP	Creation time
MEMORY_SIZE	BIGINT	Memory size of loaded task
TASK_TYPE	NVARCHAR(64)	Type of task ('PLAN' or 'PROCEDURE'), based on how the task was created
PLAN_VERSION	NVARCHAR(32)	Version of the task plan
PLAN	NCLOB	Task plan used to define the task, or task plan generated to call the procedure
COMMENTS	NVARCHAR(256)	Description of the task, from the task plan
HAS_TABLE_TYPE_INPUT	VARCHAR(5)	'TRUE' if the task is modeled with a table type as input, meaning data would need to be passed at execution time
HAS SDQ	VARCHAR(5)	'TRUE' if the task contains SDQ (smart data quality) functionality
IS_REALTIME_TASK	VARCHAR(5)	'TRUE' if the task is a realtime task, else 'FALSE'
IS_VALID	VARCHAR(5)	'TRUE' if the task is in a valid state; 'FALSE if it has been invalidated by a dependency
IS_READ_ONLY	VARCHAR(5)	'TRUE' if the task is read only (has only table type outputs), 'FALSE' if it writes to non-table-type outputs

Column	Data type	Description
PROCEDURE_SCHEMA	NVARCHAR(256)	If the task was created with a procedure instead of a plan, this attribute will contain the schema name of the stored procedure
PROCEDURE_NAME	NVARCHAR(256)	If the task was created with a procedure instead of a plan, this attribute will contain the name of the name of the stored procedure
INPUT_PARAMETER_COUNT	SMALLINT	Number of input (tableType) parameters
OUTPUT_PARAMETER	SMALLINT	Number of output (tableType) parameters
SQL_SECURITY	VARCHAR(7)	Security model for the task, either 'DEFINER' or 'INVOKER'

8.3.30 VIRTUAL_COLUMN_PROPERTIES System View [Smart Data Integration]

Lists the properties of the columns in a virtual table sent by the adapter via CREATE VIRTUAL TABLE SQL statement.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Schema name of virtual table
TABLE_NAME	NVARCHAR(256)	Virtual table name
COLUMN_NAME	NVARCHAR(256)	Virtual table column name
PROPERTY	NVARCHAR(256)	Property name
VALUE	NVARCHAR(512)	Property value

8.3.31 VIRTUAL_TABLE_PROPERTIES System View [Smart Data Integration]

Lists the properties of a virtual table sent by the adapter via the CREATE VIRTUAL TABLE SQL statement.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Schema name of virtual table
TABLE_NAME	NVARCHAR(256)	Virtual table name
PROPERTY	NVARCHAR(256)	Property name
VALUE	NCLOB	Property value. For example: • Large XSD of size 1M

8.3.32 BEST_RECORD_GROUP_MASTER_STATISTICS System View [Smart Data Quality]

Contains a summary of Best Record group master statistics.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
NUM_RECORDS	BIGINT	Total number of records processed
NUM_GROUP_MASTERS	BIGINT	Number of group master records processed
NUM_DUPLICATES	BIGINT	Number of duplicate records processed
NUM_SURVIVORS	BIGINT	Number of surviving records processed
NUM_NON_MATCH_RECORDS	BIGINT	Number of non-matching records processed

8.3.33 BEST_RECORD_RESULTS System View [Smart Data Quality]

Contains governance information for every column in every record that is updated in the best record process.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
DST_TABLE_NAME	NVARCHAR(128)	Name of the destination table for the operation
DST_ROW_ID	BIGINT	Unique identifier for the destination row
DST_COLUMN_NAME	NVARCHAR(128)	Name of the destination column in the destination table
DST_ROW_TYPE	NVARCHAR(1)	Identifies how the record was updated or if it was newly created
SRC_TABLE_NAME	NVARCHAR(128)	Name of the source table for the operation
SRC_ROW_ID	BIGINT	Unique identifier for the source row
SRC_COLUMN_NAME	NVARCHAR(128)	Name of the source column in the source table
STRATEGY_GROUP_ID	INTEGER	Identification number that identifies the best record strategy group
STRATEGY_ID	INTEGER	Identification number that identifies each strategy listed in the strategy group
BEST_RECORD_RULE	NVARCHAR(256)	Name of the rule that updates one or more columns as it is defined in the best record configuration
ACTION_NAME	NVARCHAR(256)	Name of the action that updates a column as it is defined in the best record configuration
UPDATE_NUM	INTEGER	Number of times the column was updated in the best record process

Column	Data type	Description
OPERATION_TYPE	NVARCHAR(1)	Identifies how the record was updated in the best record process

8.3.34 BEST_RECORD_STRATEGIES System View [Smart Data Quality]

Contains information on which strategies are used in each strategy group and in which order.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
STRATEGY_GROUP_NAME	NVARCHAR(256)	Name of the strategy group as defined in the best record configuration
STRATEGY_ID	INTEGER	Identification number that identifies each strategy listed in the strategy group
STRATEGY_ORDER	INTEGER)	Order of the strategy as defined in the list of strategies
STRATEGY_NAME	NVARCHAR(256)	Name of the strategy as defined in the best record configuration

8.3.35 CLEANSE_ADDRESS_RECORD_INFO System View [Smart Data Quality]

Describes how well an address was assigned as well as the type of address.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
TABLE_NAME	NVARCHAR(128)	Name of the table defined in the task plan for the operation
ROW_ID	BIGINT	Unique identifier of the row processed for this execution of the task plan
ENTITY_INSTANCE	INTEGER	Identifier to differentiate between multiple entities processed in a row
ENTITY_INSTANCE_OCCURRENCE	INTEGER	Unique identifier to identify the occurrence of an entity
DATA_SOURCE	NVARCHAR(256)	Source where the data was produced
ISO_COUNTRY_2CHAR	NVARCHAR(4)	Two-character country code
ASSIGNMENT_TYPE	NVARCHAR(4)	Code that represents the type of an address
ASSIGNMENT_INFORMATION	NVARCHAR(4)	Code that specifies the validity of an address
ASSIGNMENT_LEVEL	NVARCHAR(4)	Code that represents the level to which the address matched data in the address reference data

8.3.36 CLEANSE_CHANGE_INFO System View [Smart Data Quality]

Describes the changes made during the cleansing process.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
TABLE_NAME	NVARCHAR(128)	Name of the table defined in the task plan for the operation
ROW_ID	BIGINT	Unique identifier of the row processed for this execution of the task plan
ENTITY_ID	NVARCHAR(12)	Identifier describing the type of record that was processed
ENTITY_INSTANCE	INTEGER	Identifier to differentiate between multiple entities processed in a row
ENTITY_INSTANCE_OCCURRENCE	INTEGER	Unique identifier to identify the occurrence of an entity
COMPONENT_ID	NVARCHAR(12)	Identification number that refers to data components
COMPONENT_ELEMENT_ID	NVARCHAR(12)	Identification number that refers to more granular elements within a component
DATA_SOURCE	NVARCHAR(256)	Source where the data was produced
CHANGE_SIGNIFICANCE_ID	NVARCHAR(12)	Identification number that refers to the significance of the change

8.3.37 CLEANSE_COMPONENT_INFO System View [Smart Data Quality]

Identifies the location of parsed data elements in the input and output.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
ENTITY_ID	NVARCHAR(12)	Identifier describing a data attribute such as a person name, organization name, address and so on.
ENTITY_INSTANCE	INTEGER	Identifier to differentiate between multiple entities processed in a row
ENTITY_INSTANCE_OCCURRENCE	INTEGER	Unique identifier to identify the occurrence of an entity
DATA_SOURCE	NVARCHAR(256)	Source where the data originated
COMPONENT_ID	NVARCHAR(12)	Identification number that refers to data components
COMPONENT_ELEMENT_ID	NVARCHAR(12)	Identification number that refers to more granular elements within a component
TABLE_NAME	NVARCHAR(128)	Name of the input table where the component element was found
ROW_ID	BIGINT	Unique identifier of the row processed for this execution of the task plan
COLUMN_NAME	NVARCHAR(128)	Name of the column in the input table where the component element was found
COLUMN_START_POSITION	INTEGER	Starting character of the component element in the input column
COLUMN_DATA_LENGTH	INTEGER	Number of characters of the component element in the input column
OUTPUT_TABLE_NAME	NVARCHAR(128)	Name of the output table where the component element was written
OUTPUT_COLUMN_NAME	NVARCHAR(128)	Name of the column in the output table where the component element was written

Column	Data type	Description
OUTPUT_COLUMN_START_POSITION	INTEGER	Starting character of the component element in the output column
OUTPUT_COLUMN_DATA_LENGTH	INTEGER	Number of characters of the component element in the output column

8.3.38 CLEANSE_INFO_CODES System View [Smart Data Quality]

Contains one row per info code generated by the cleansing process.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
TABLE_NAME	NVARCHAR(128)	Name of the table defined in the task plan for the operation
ROW_ID	BIGINT	Unique identifier of the row processed for this execution of the task plan
ENTITY_ID	NVARCHAR(12)	Identifier describing the type of record that was processed
ENTITY_INSTANCE	INTEGER	Identifier to differentiate between multiple entities processed in a row
ENTITY_INSTANCE_OCCURRENCE	INTEGER	Unique identifier to identify the occurrence of an entity
DATA_SOURCE	NVARCHAR(256)	Source where the data was produced
INFO_CODE	NVARCHAR(10)	Information code that gives information about the processing of the record

8.3.39 CLEANSE_STATISTICS System View [Smart Data Quality]

Contains a summary of Cleanse statistics.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
ENTITY_ID	NVARCHAR(12)	Identifier describing the type of record that was processed
ENTITY_INSTANCE	INTEGER	Identifier to differentiate between multiple entities processed in a row
NUM_RECORDS	BIGINT	Total number of records processed for the entity instance
NUM_VALIDS	BIGINT	Number of valid records processed for the entity instance
NUM_SUSPECTS	BIGINT	Number of suspect records processed for the entity instance
NUM_BLANKS	BIGINT	Number of blank records processed for the entity instance
NUM_HIGH_SIGNIFICANT_CHANGES	BIGINT	Number of records with high significance changes for the entity instance

8.3.40 GEOCODE_INFO_CODES System View [Smart Data Quality]

Contains one row per info code generated by the geocode transformation process.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
TABLE_NAME	NVARCHAR(128)	Name of the table defined in the task plan for the operation
ROW_ID	BIGINT	Unique identifier of the row processed for this execution of the task plan
DATA_SOURCE	NVARCHAR(256)	Source where the data was produced
INFO_CODE	NVARCHAR(10)	Information code generated by the geocode transformation operation

8.3.41 GEOCODE_STATISTICS System View [Smart Data Quality]

Contains a summary of Geocode statistics.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
NUM_RECORDS	BIGINT	Total number of records processed
NUM_ASSIGNED	BIGINT	Number of assigned records processed

Column	Data type	Description
NUM_UNASSIGNED	BIGINT	Number of unassigned records processed

8.3.42 MATCH_GROUP_INFO System View [Smart Data Quality]

Contains one row for each match group.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
GROUP_ID	INTEGER	Group identification number
GROUP_COUNT	INTEGER	Number of records in the match group
SOURCE_COUNT	INTEGER	Number of sources represented in the match group
REVIEW_GROUP	NVARCHAR(1)	Indicates whether the group is flagged for review
CONFLICT_GROUP	NVARCHAR(1)	Indicates whether the group is flagged for conflict

8.3.43 MATCH_RECORD_INFO System View [Smart Data Quality]

Contains one row for each matching record per level.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
TABLE_NAME	NVARCHAR(128)	Name of the table defined in the task plan for the operation
ROW_ID	BIGINT	Unique identifier of the row processed for this execution of the task plan
GROUP_ID	INTEGER	Group identification number

8.3.44 MATCH_SOURCE_STATISTICS System View [Smart Data Quality]

Contains counts of matches within and between data sources.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
SOURCE_NAME	NVARCHAR(256)	Data source name
RELATED_SOURCE_NAME	NVARCHAR(256)	Related data source name

Column	Data type	Description
NUM_MATCH_DECISIONS	INTEGER	Number of comparisons resulting in a match decision between records in each SOURCE_ID/RELATED_SOURCE_ID pair

8.3.45 MATCH_STATISTICS System View [Smart Data Quality]

Contains statistics regarding the run of the transformation operation.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
NUM_RECORDS	BIGINT	Total number of records processed by the transformation operation
NUM_MATCH_RECORDS	BIGINT	Number of records that reside in match groups
NUM_NON_MATCH_RECORDS	BIGINT	Number of non-matching records that do not reside in match groups
NUM_MATCH_GROUPS	BIGINT	Number of match groups identified
NUM_REVIEW_GROUPS	BIGINT	Number of match groups flagged for review
NUM_NON_REVIEW_GROUPS	BIGINT	Number of match groups not flagged for review
NUM_CONFLICT_GROUPS	BIGINT	Number of match groups flaggged with conflicts
NUM_COMPARISONS_PERFORMED	BIGINT	Number of comparisons performed by the transformation operation

Column	Data type	Description
NUM_MATCH_DECISIONS	BIGINT	Number of comparisons resulting in a match decision

8.3.46 MATCH_TRACING System View [Smart Data Quality]

Contains one row for each match decision made during the matching process.

Structure

Column	Data type	Description
SCHEMA_NAME	NVARCHAR(256)	Name of the schema where the task is located
TASK_NAME	NVARCHAR(256)	Name of the task
TASK_EXECUTION_ID	BIGINT	Unique identifier for a particular run of a task plan created when START TASK is called
OPERATION_NAME	NVARCHAR(128)	Name of the operation in the task plan
TABLE_NAME	NVARCHAR(128)	Name of the table defined in the task plan for the operation
ROW_ID	BIGINT	Unique identifier of the row processed for this execution of the task plan
RELATED_TABLE_NAME	NVARCHAR(128)	Name of the table defined in the task plan for an operation
RELATED_ROW_ID	BIGINT	Unique identifier of the row processed for this execution of the task plan
POLICY_NAME	NVARCHAR(256)	Name of the match policy that processed the related rows
RULE_NAME	NVARCHAR(256)	Name of the match rule that processed the related rows
SCORE	INTEGER	Similarity score of the related rows

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

www.sap.com/contactsap

© 2026 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see https://www.sap.com/about/legal/trademark.html for additional trademark information and notices.