SAP HANA Administration Guide
Table of Contents

1 Getting Started. ................................................................. 8
  1.1 About This Document. .................................................... 8
  1.2 Related Information. ..................................................... 8
  1.3 Important SAP Notes. .................................................... 9
  1.4 Technical System Landscape. ........................................... 12

2 Working with the SAP HANA Studio. ..................................... 14
  2.1 About the SAP HANA Studio. ........................................... 14
  2.2 Updating the SAP HANA Studio. ....................................... 14
  2.3 Screen Areas of the SAP HANA Studio. ............................ 15
  2.4 Opening the Administration Console of the SAP HANA Studio. 16
  2.5 Opening the Administration Editor. ................................... 17
  2.6 Opening the SQL Console. ............................................ 18
  2.7 Executing SQL Statements. ............................................ 18
  2.8 Customizing the Administration Console. ......................... 19

3 Managing SAP HANA Systems in the SAP HANA Studio. ............. 24
  3.1 Adding SAP HANA Systems in the SAP HANA Studio. ............. 24
  3.2 Adding Users to an SAP HANA System. ................................ 25
  3.3 Organizing SAP HANA Systems Using Folders. .................... 26
  3.4 Exporting List of SAP HANA Systems. ................................ 26
  3.5 Importing List of SAP HANA Systems. ............................... 27
  3.6 Searching for SAP HANA Systems. ................................... 27

4 Starting and Stopping SAP HANA Systems. .............................. 28
  4.1 Starting SAP HANA Systems. .......................................... 28
  4.2 Stopping SAP HANA Systems. ......................................... 28
  4.3 Restarting SAP HANA Systems. ....................................... 29
  4.4 Stopping and Starting Database Services. ......................... 30
  4.5 Monitoring SAP HANA Systems During Stop and Start. .......... 31

5 Configuring SAP HANA System Properties. ............................. 32
  5.1 Changing System Properties. .......................................... 32
  5.2 Resetting System Properties. ......................................... 32
  5.3 Setting the global_allocation_limit Parameter. .................. 33
  5.4 Changing the Default SLD Data Supplier Configuration. ........... 34
     5.4.1 System Landscape Directory – Additional Resources. ...... 35
  5.5 Configuring Trace File Rotation. ...................................... 36
6 Managing SAP HANA Licenses............................................................................. 38
6.1 License Types................................................................................................. 38
6.2 Checking the Current License Key................................................................. 39
6.3 Installing Permanent Licenses........................................................................ 40
6.4 Deleting Existing Permanent License Keys..................................................... 41

7 Managing SAP HANA Users............................................................................. 42
7.1 Database Users............................................................................................... 42
7.2 Operating System User.................................................................................... 42
7.3 User Authentication.......................................................................................... 43
  7.3.1 Working with the Password Blacklist.......................................................... 43
  7.3.2 Configuring Kerberos for SAP HANA Database Hosts............................... 44
  7.3.3 Configuring SAML Identity Providers........................................................ 45
  7.3.4 Problems with User Authentication.............................................................. 46
7.4 User Authorization............................................................................................ 48
  7.4.1 Privileges for Administration Tasks.............................................................. 49
7.5 User Provisioning............................................................................................. 50
  7.5.1 Roles........................................................................................................... 50
  7.5.2 Creating Roles in Runtime........................................................................... 52
  7.5.3 Changing Roles in Runtime......................................................................... 53
  7.5.4 Creating Users............................................................................................. 53
  7.5.5 Changing Users........................................................................................... 54
  7.5.6 Deleting Users............................................................................................. 55
  7.5.7 Deactivating Users...................................................................................... 56
  7.5.8 Reactivating Users...................................................................................... 56
  7.5.9 Verifying Users’ Privileges......................................................................... 56
  7.5.10 Prerequisites for Granting and Revoking Privileges and Roles............... 57
7.6 Disabling Default User Filtering of Schemas.................................................. 58

8 Monitoring SAP HANA Systems...................................................................... 60
8.1 The System Monitor........................................................................................ 60
8.2 Monitoring in the Administration Editor......................................................... 61
8.3 Monitoring Overall System Status................................................................. 62
8.4 Monitoring System Status and Resource Usage............................................. 65
8.5 Monitoring Host Status and Failover Configuration....................................... 68
8.6 Monitoring Alerts............................................................................................. 73
  8.6.1 Statistics Server Checks.............................................................................. 75
  8.6.2 Configuring E-Mail Notifications for Alerts............................................... 82
  8.6.3 Configuring Check Thresholds..................................................................... 83
  8.6.4 Configuring Start Times of Periodic Checks.............................................. 84
8.7 Monitoring System Performance..................................................................... 84
  8.7.1 Additional Performance Monitoring Tools................................................ 86
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.8</td>
<td>Monitoring Disk Space</td>
<td>88</td>
</tr>
<tr>
<td>8.8.1</td>
<td>About Persistent Data Storage in the SAP HANA Database</td>
<td>90</td>
</tr>
<tr>
<td>8.8.2</td>
<td>Resolving Disk-Full Events</td>
<td>92</td>
</tr>
<tr>
<td>8.9</td>
<td>Monitoring Memory Usage</td>
<td>92</td>
</tr>
<tr>
<td>8.9.1</td>
<td>What Is Memory Used For?</td>
<td>93</td>
</tr>
<tr>
<td>8.9.2</td>
<td>SAP HANA Used Memory and Peak Used Memory</td>
<td>94</td>
</tr>
<tr>
<td>8.9.3</td>
<td>Memory Usage of Column and Row Tables</td>
<td>95</td>
</tr>
<tr>
<td>8.9.4</td>
<td>Allocated Memory Pools and Allocation Limits</td>
<td>97</td>
</tr>
<tr>
<td>8.9.5</td>
<td>SAP HANA Memory Usage and the Operating System</td>
<td>98</td>
</tr>
<tr>
<td>9</td>
<td>Managing Tables in the SAP HANA Database</td>
<td>102</td>
</tr>
<tr>
<td>9.1</td>
<td>Column-Based and Row-Based Storage in the SAP HANA Database</td>
<td>102</td>
</tr>
<tr>
<td>9.1.1</td>
<td>Creating Tables</td>
<td>103</td>
</tr>
<tr>
<td>9.1.2</td>
<td>Creating Views</td>
<td>103</td>
</tr>
<tr>
<td>9.2</td>
<td>Memory Management in the Column Store</td>
<td>104</td>
</tr>
<tr>
<td>9.2.1</td>
<td>Loading and Unloading Column Tables into and from Memory</td>
<td>105</td>
</tr>
<tr>
<td>9.3</td>
<td>The Delta Merge Operation</td>
<td>107</td>
</tr>
<tr>
<td>9.3.1</td>
<td>Merge Motivations</td>
<td>108</td>
</tr>
<tr>
<td>9.3.2</td>
<td>The Merge Monitor</td>
<td>110</td>
</tr>
<tr>
<td>9.3.3</td>
<td>Cost Functions</td>
<td>111</td>
</tr>
<tr>
<td>9.3.4</td>
<td>Merge Tokens</td>
<td>113</td>
</tr>
<tr>
<td>9.3.5</td>
<td>Monitoring Delta Merge History</td>
<td>114</td>
</tr>
<tr>
<td>9.3.6</td>
<td>Performing a Manual Delta Merge Operation</td>
<td>115</td>
</tr>
<tr>
<td>9.3.7</td>
<td>Viewing Progress of Delta Merge Operations</td>
<td>116</td>
</tr>
<tr>
<td>9.4</td>
<td>Data Compression in the Column Store</td>
<td>117</td>
</tr>
<tr>
<td>9.4.1</td>
<td>Checking Compression of Column Tables</td>
<td>118</td>
</tr>
<tr>
<td>9.4.2</td>
<td>Compressing Column Tables Manually</td>
<td>119</td>
</tr>
<tr>
<td>9.5</td>
<td>Opening Tables in the Table Editor</td>
<td>120</td>
</tr>
<tr>
<td>9.5.1</td>
<td>Table Editor Views for Individual Table Analysis</td>
<td>121</td>
</tr>
<tr>
<td>9.6</td>
<td>Exporting Tables and Other Catalog Objects</td>
<td>123</td>
</tr>
<tr>
<td>9.7</td>
<td>Importing Tables and Other Catalog Objects</td>
<td>124</td>
</tr>
<tr>
<td>10</td>
<td>Scaling SAP HANA</td>
<td>127</td>
</tr>
<tr>
<td>10.1</td>
<td>About Scalability</td>
<td>127</td>
</tr>
<tr>
<td>10.2</td>
<td>Network Configuration</td>
<td>128</td>
</tr>
<tr>
<td>10.3</td>
<td>Perform a Distributed SAP HANA Setup</td>
<td>130</td>
</tr>
<tr>
<td>10.4</td>
<td>Call Options for Adding/Removing Hosts</td>
<td>131</td>
</tr>
<tr>
<td>10.5</td>
<td>Starting and Stopping a Distributed SAP HANA System</td>
<td>132</td>
</tr>
<tr>
<td>10.6</td>
<td>Table Distribution in SAP HANA</td>
<td>132</td>
</tr>
<tr>
<td>10.7</td>
<td>Table Partitioning in the SAP HANA Database</td>
<td>134</td>
</tr>
<tr>
<td>10.7.1</td>
<td>Single-Level Partitioning</td>
<td>135</td>
</tr>
<tr>
<td>10.7.2</td>
<td>Multi-Level Partitioning</td>
<td>137</td>
</tr>
</tbody>
</table>
13.8 Diagnostics Files for Backup and Recovery. .................................................... 175
13.9 Backing Up the SAP HANA Database. ............................................................. 176
  13.9.1 Performing a Data Backup Using SAP HANA Studio. .......................... 176
  13.9.2 Performing a Data Backup Using SQL Commands. .............................. 177
  13.9.3 Performing a Data Backup in Batch Mode. .......................................... 178
  13.9.4 Canceling a Running Data Backup. ...................................................... 178
13.10 About SAP HANA Database Recovery. ...................................................... 179
  13.10.1 Recovery Scenarios. ............................................................................. 179
13.11 Recovering the SAP HANA Database. .......................................................... 181
13.12 Copying a Database Using Backup and Recovery. ..................................... 184
13.13 Backup Checks. ......................................................................................... 185
13.14 SQL Syntax for Backup and Recovery. ....................................................... 187
  13.14.1 SQL Statements for Data Backup. ......................................................... 188
  13.14.2 SQL Statement for Data Backup Check. .............................................. 190
  13.14.3 SQL Statements for Data Backup Cancel. .......................................... 191
  13.14.4 SQL Statements for Data Snapshot. ..................................................... 191
  13.14.5 SQL Statements for Database Recovery and Data Recovery. ............. 192
14 High Availability and SAP HANA. ................................................................. 196
  14.1 About High Availability. .......................................................................... 196
    14.1.1 SAP HANA Disaster Recovery Support. ............................................ 197
    14.1.2 SAP HANA Fault Recovery Support. ................................................ 198
  14.2 Setting up System Replication. ................................................................ 199
    14.2.1 Configuring the Primary System. ....................................................... 200
    14.2.2 Configuring the Secondary System. .................................................... 201
    14.2.3 Performing a Takeover. ..................................................................... 202
    14.2.4 Performing a Failback. ..................................................................... 203
    14.2.5 Example of Setting up System Replication. ....................................... 203
    14.2.6 System Replication Configuration Parameters. ............................... 204
    14.2.7 Monitoring System Replication. ........................................................ 205
    14.2.8 Adding a New Host to a Replicated System. .................................... 206
  14.3 Setting up Host Auto-Failover. .................................................................. 207
    14.3.1 Configuring Clients for Failover. ....................................................... 207
15 Query Analysis in the SAP HANA Database. .............................................. 209
  15.1 Analyzing SQL Execution with the Plan Explanation. ............................. 209
  15.2 Analyzing SQL Execution with the Plan Visualizer. ................................ 209
  15.3 Analyzing SQL Execution with the SQL Plan Cache. .............................. 213
  15.4 Example: Reading the SQL Plan Cache. ................................................... 214
16 Getting Support. ............................................................................................ 219
  16.1 Opening a Support Connection. ................................................................. 219
1 Getting Started

1.1 About This Document

The SAP HANA Administration Guide describes the main tasks and concepts necessary for the ongoing operation of SAP HANA.

The following areas are covered:

- SAP HANA studio as an SAP HANA administration tool
- Starting and stopping SAP HANA systems
- System configuration
- License management
- User provisioning
- Monitoring (for example, performance, memory usage, disk usage, alert situations)
- Backup and recovery
- Distributed system management
- High availability
- Other administrative tasks, for example, managing tables, managing persistence encryption, auditing database activity, and so on.

For more information about the SAP HANA landscape, security, and installation, see SAP Help Portal at http://help.sap.com/hana_appliance.

1.2 Related Information

For more information about planning your deployment, see the following:

<table>
<thead>
<tr>
<th>Content</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latest documentation for <em>SAP HANA</em></td>
<td><a href="https://service.sap.com/hana">https://service.sap.com/hana</a>*</td>
</tr>
<tr>
<td></td>
<td><a href="http://help.sap.com/hana_appliance">http://help.sap.com/hana_appliance</a></td>
</tr>
<tr>
<td>Sybase product documentation</td>
<td><a href="http://sybooks.sybase.com/nav/base.do">http://sybooks.sybase.com/nav/base.do</a></td>
</tr>
<tr>
<td>Latest versions of SAP installation and upgrade guides</td>
<td><a href="https://service.sap.com/instguides">https://service.sap.com/instguides</a></td>
</tr>
<tr>
<td>Sizing, calculation of hardware requirements, such as CPU,</td>
<td><a href="https://service.sap.com/sizing">https://service.sap.com/sizing</a></td>
</tr>
<tr>
<td>disk, and memory resources</td>
<td></td>
</tr>
<tr>
<td>SAP BusinessObjects installation and deployment guides</td>
<td><a href="https://service.sap.com/bosap-instguides">https://service.sap.com/bosap-instguides</a></td>
</tr>
</tbody>
</table>
SAP BusinessObjects support
https://service.sap.com/bosap-support

Sizing, calculation of hardware requirements - such as CPU, disk and memory resources - with the QuickSizer tool
https://service.sap.com/quicksizer

Released platforms and technology-related topics such as maintenance strategies and language support – Platform Availability Matrix (PAM)
https://service.sap.com/pam

Network security
https://service.sap.com/securityguide

High availability
https://service.sap.com/ha

Performance
https://service.sap.com/performance

Security
https://service.sap.com/security

Unicode SAP systems and their availability
https://service.sap.com/unicode

Information about Support Package Stacks, latest software versions and patch level requirements
https://service.sap.com/sp-stacks

(*) You need an authorized user ID to access this information. Do you need to register for an SAP Service Marketplace login? Did you forget your password?

The following table lists further useful links on SAP Service Marketplace:

<table>
<thead>
<tr>
<th>Content</th>
<th>Location on SAP Service Marketplace</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP message wizard</td>
<td><a href="https://service.sap.com/message">https://service.sap.com/message</a></td>
</tr>
<tr>
<td>SAP Notes search</td>
<td><a href="https://service.sap.com/notes">https://service.sap.com/notes</a></td>
</tr>
<tr>
<td>SAP Software Distribution Center – software download and ordering of software</td>
<td><a href="https://service.sap.com/swdc">https://service.sap.com/swdc</a></td>
</tr>
<tr>
<td>SAP Online Knowledge Products (OKPs) – role-specific learning maps</td>
<td><a href="https://service.sap.com/rkt">https://service.sap.com/rkt</a></td>
</tr>
</tbody>
</table>

1.3 Important SAP Notes

You must read the following SAP Notes before you start the installation. These SAP Notes contain the latest information about the installation, as well as corrections to the installation documentation.

Make sure that you have the most up-to-date version of each SAP Note, which you can find on SAP Service Marketplace at https://service.sap.com/notes.
<table>
<thead>
<tr>
<th>SAP Note Number</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1514967</td>
<td>SAP HANA: Central Note</td>
</tr>
<tr>
<td>1771591</td>
<td>SAP HANA Appliance Software SPS 05 Release Note</td>
</tr>
<tr>
<td>1523337</td>
<td>SAP HANA Database: Central Note</td>
</tr>
<tr>
<td>1681092</td>
<td>Support for multiple SAP HANA databases on a single SAP HANA appliance</td>
</tr>
<tr>
<td>1661202</td>
<td>Support for multiple applications on SAP HANA</td>
</tr>
<tr>
<td>1577128</td>
<td>Supported clients for SAP HANA</td>
</tr>
<tr>
<td>1514966</td>
<td>SAP HANA: Sizing SAP HANA Database</td>
</tr>
<tr>
<td>1637145</td>
<td>SAP BW on HANA: Sizing SAP HANA Database</td>
</tr>
<tr>
<td>1597355</td>
<td>Swap space recommendation for Linux</td>
</tr>
</tbody>
</table>

For the SAP HANA system there are no special requirements regarding swap space.

Check the current SAP Notes for the various parts of SAP HANA by searching for any of the following application areas:

**SAP HANA Platform Edition**

- **BC-DB-HDB** SAP HANA Database
- **BC-DB-HDB-AFL** SAP HANA Advanced Function Library
- **BC-DB-HDB-AFL-PAL** SAP HANA Predictive Analysis Library
- **BC-DB-HDB-AFL-SOP** SAP HANA Sales and & Operations Planning
- **BC-DB-HDB-BAC** SAP HANA Backup & Recovery
- **BC-DB-HDB-CCM** CCMS for SAP HANA
- **BC-DB-HDB-CLI** SAP HANA Clients (JDBC, ODBC)
- **BC-DB-HDB-DBA** SAP HANA database administration (DBA cockpit)
- **BC-DB-HDB-DXC** SAP HANA direct extractor connector
- **BC-DB-HDB-ENG** SAP HANA database engine
- **BC-DB-HDB-MDX** MDX Engine / MS Excel client
- **BC-DB-HDB-PER** SAP HANA database persistence
- **BC-DB-HDB-PLE** Planning Engine
- **BC-DB-HDB-POR** DB Porting for SAP HANA
- **BC-DB-HDB-R** SAP HANA Integration with R
- **BC-DB-HDB-SCR** SAP HANA SQL Script
- **BC-DB-HDB-SEC** SAP HANA Security & User Management
- **BC-DB-HDB-SYS** SAP HANA database interface/DBMS
- **BC-DB-HDB-TXT** SAP HANA Text & Search Features
- **BC-DB-HDB-XS** SAP HANA Application Services
- **BC-HAN-MOD** SAP HANA studio / information modeler
- **BC-HAN-3DM** SAP HANA information composer
- **BC-HAN-SRC** SAP HANA UI Toolkit
- **BC-CCM-HAG** SAP Host Agent
- **SV-SMG-DIA** SAP Solution Manager Diagnostics Agent

**SAP HANA Lifecycle Management**

- **BC-HAN-SL-STP** SAP HANA unified installer
- **BC-DB-HDB-INS** SAP HANA database installation (hdbinst)
- **BC-HAN-UPD** Software Update Manager for SAP HANA
- **BC-DB-HDB-UPG** SAP HANA database upgrade (hdbupd)

**SAP HANA Enterprise Edition**

- **BC-HAN-LOA** SAP HANA load controller (log-based replication)
- **BC-HAN-REP** Sybase rep. server (log-based replication)
- **BC-HAN-LTR** SAP Landscape Transformation (trigger-based replication)
- **BC-HAN-DXC** SAP HANA Direct Extractor Connection
- **EIM-DS** Data Services (ETL-based replication)

**End User Clients**

- **BI-BIP, BI-BIP-CMC** Business intelligence platform (formerly known as BOE)
- **BI-RA-EXP** SAP BusinessObjects Explorer
- **BI-RA-CR, BI-BIP-CRS** SAP Crystal Reports
- **BI-RA-XL** Dashboard Designer
- **BI-BIP-IDT** Information design tool
- **BI-RA-WBI** Web Intelligence
- **BI-RA-AO-XLA** MS Excel Add-In

The search also supports using the wildcard asterisk (*) in searches, so you can, for example, also search for BC-DB-HDB* or similar and you will get results for all sub-components.

**Customer Messages**

If you encounter any problems with the software, create a customer message on the SAP Service Marketplace at [http://service.sap.com/message](http://service.sap.com/message).

In addition the Customer Interaction Center (CIC) is available 24 x 7 in every region to help you resolve any issues you may run into ([https://service.sap.com/supportcenters](https://service.sap.com/supportcenters)).

The CIC requires a valid S-user number. Follow the steps in this guide to create an S-user ID ([https://service.sap.com/~sapidp/011000358700000905192010E.pdf](https://service.sap.com/~sapidp/011000358700000905192010E.pdf)).

While creating a customer message, you can chose from the above list of components for the relevant software part.
1.4 Technical System Landscape

The figure below shows an overview of the technical system landscape for the SAP HANA appliance software and its related components, such as the SAP HANA studio and other applications, one of which is the SAP HANA information composer. Note that the figure below shows a sample configuration with three SAP HANA database servers. The figure also shows some optional components that may be purchased separately.
## 2 Working with the SAP HANA Studio

### 2.1 About the SAP HANA Studio

The SAP HANA studio runs on the Eclipse platform and is both the central development environment and the main administration tool for SAP HANA.

Administrators use the SAP HANA studio, for example, to start and stop services, to monitor the system, to configure system settings, and to manage users and authorizations. The SAP HANA studio accesses the servers of the SAP HANA database by SQL. Developers can use the SAP HANA studio to create content such as modeled views and stored procedures. These development artifacts are stored in the repository, which is part of the SAP HANA database. The SAP HANA studio is developed in Java and based on the Eclipse platform.

The SAP HANA studio presents its various tools in the form of perspectives. Database administration and monitoring features are contained primarily within the Administration Console perspective. Additional perspectives include the Modeler perspective, the SAP HANA Development, and the Lifecycle Management perspective.

### 2.2 Updating the SAP HANA Studio

To ensure that you are working with the most recent version of the SAP HANA studio, you need to check regularly for updates.

Before you can update the SAP HANA studio manually and configure the SAP HANA studio to check automatically for updates, you must have configured the site from which updates are downloaded.

1. In the SAP HANA studio, specify the update site as follows:
   a) From the main menu, choose **Window** ➤ **Preferences** ➤ **Install/Update** ➤ **Available Software Sites**
   b) Choose **Add...** and specify the name of the update repository (optional) and its location, for example, http://<host_name>:<port_number>/tools/hdb.studio.update or file://///update_server/hdbstudio/repository/.

2. To update the SAP HANA studio manually, proceed as follows:
   a) From the main menu, choose **Help** ➤ **Check for Updates**
      The SAP HANA studio checks the specified software site for an update.
   b) If an update is available, follow the on-screen instructions to install the update.

3. To configure the SAP HANA studio to check for updates automatically and notify you of their availability, proceed as follows:
   a) From the main menu, choose **Window** ➤ **Preferences** ➤ **Install/Update** ➤ **Automatic Updates**
   b) Specify your update settings.
      You are automatically notified if an update is available in line with your settings.
2.3 Screen Areas of the SAP HANA Studio

The SAP HANA studio presents its tools in the form of perspectives. A perspective contains specific task- or resource-related functions. It determines which views and editors are available and controls what appears in certain menus and toolbars.

The following figure shows the screen areas of the SAP HANA studio with the Administration Console perspective open:

The following is a brief overview of the various screen areas:

<table>
<thead>
<tr>
<th>Screen Area</th>
<th>Description</th>
</tr>
</thead>
</table>
| Main menu and main toolbar| The main menu contains standard Eclipse functions. The main toolbar is located beneath the main menu. The contents of this toolbar change based on the active perspective. Items in the toolbar might be enabled or disabled based on the state of either the active view or editor. The perspective switcher is an additional toolbar normally
|"Navigator" view         |             |
|Editor area (for example, Administration editor, SQL console, table editor)|           |
|Other views (for example, Error Log and Properties)|             |
Screen Area | Description
--- | ---
 | located on the top-right of the screen, next to the main toolbar. It allows quick access to perspectives that are currently open. It also has a button that can open new perspectives.

Editor area

Each perspective has editors for editing and browsing resources. Editors are displayed as tabs in the editor area. Several editors can be open at the same time.

The Administration editor and SQL console, for example, are available in the **Administration Console** perspective.

Views

Views support editors and provide alternative presentations as well as ways to navigate the information in the SAP HANA studio. For example, the **Error Log** contains error and information messages.

To open a view, from the main menu, choose ***Window*** ‣ **Show View***.

Navigator view

The **Navigator** is a view that provides you with a list of all SAP HANA systems managed in the SAP HANA studio and allows you to drill down easily into each system. The **Navigator** view toolbar contains icons that provide you with quick access to several editors and functions, including:

- System Monitor
- Administration editor
- SQL console
- **Find System** and **Find Table** search functions

You can filter the entries in several nodes of the navigator by right-clicking the node and choosing **Filters...**

**Related Links**

*Eclipse Workbench User Guide*

---

### 2.4 Opening the Administration Console of the SAP HANA Studio

To access the database administration and monitoring features of the SAP HANA studio, you open the Administration Console perspective.

1. From your file explorer, start **hdbstudio.exe**.
2. On the **Welcome** page, choose **Open Administration Console**.

The Administration Console opens.

**Note:** Once you have closed the **Welcome** page, you can always change from another perspective to the Administration Console perspective by choosing ***Window*** ‣ **Open Perspective** ‣ **Administration Console**
or by choosing the Administration Console button in the perspective switcher in the upper-right corner of the screen.

Related Links
Customizing the Administration Console [page 19]
There are many options available for customizing the Administration Console of the SAP HANA studio.

2.5 Opening the Administration Editor

The Administration editor is available in the Administration Console perspective and is the main tool for performing administration and monitoring activities.

Before you can open the Administration editor, you must have added an SAP HANA system in the SAP HANA studio. You must also have system privilege CATALOG READ or DATA ADMIN.

1. Select the system in the Navigator view.
2. Open the Administration editor in one of the following ways:
   - From the Navigator view toolbar, choose the (Administration) button.
   - Double-click the system.
   - In the context menu, choose Administration.

The Administration editor appears. The header of Administration editor contains general information about the system (name, host, instance number, time of last refresh), as well a toolbar with the following functions:

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>✂️ (Refresh current page)</td>
<td>Allows you to manually refresh the tab you are viewing</td>
</tr>
<tr>
<td>⏯️ (Stop/Start automatic refresh)</td>
<td>Allows you to activate and deactivate automatic refresh</td>
</tr>
<tr>
<td></td>
<td>You can specify the interval between automatic refresh (in seconds) in the corresponding field.</td>
</tr>
<tr>
<td>📋 (Copy to clipboard)</td>
<td>Allows you to copy the details of the tab you are viewing to the clipboard and then to paste this to another program, for example Notepad</td>
</tr>
<tr>
<td>✸ (Clear messages)</td>
<td>Allows you to clear any messages displayed in the header</td>
</tr>
</tbody>
</table>

The other tabs of the Administration editor provide you with detailed information and functions for monitoring and administrating the system.

Related Links
Adding SAP HANA Systems in the SAP HANA Studio [page 24]
To work with and manage an SAP HANA system in the SAP HANA studio, you must create a connection to the system and configure communication.
2.6 Opening the SQL Console

Some tasks may require you to work with SQL statements, for example, certain administration tasks can only be performed using SQL. You can enter, execute, and analyze SQL statements in the SQL console.

1. In the Navigator view, select the system to which you want to establish a connection.
   You can also navigate to the specific catalog object that you want to execute on.

2. Open the SQL console in one of the following ways:
   - From the Navigator view toolbar, choose the button.
   - From the context menu, choose SQL Console.

   The SQL console appears with the connected system and user in the header. If you opened the SQL console from a specific catalog object, the schema is also displayed.

   To connect to a different system from within the SQL console, choose the Choose Connection button in the toolbar in the top-right of the editor and choose the catalog of another system.

   **Note:** There are several options available for customizing the behavior of SQL statement execution in the SQL console. From the main menu choose Window ➔ Preferences ➔ SQL. For more information about the options, see the section on customizing the Administration Console.

2.7 Executing SQL Statements

In the SAP HANA studio, you can execute SQL statements in the SQL console.

To execute an SQL statement, you must have the required privileges to perform the specified operation on the table or view involved.

1. Open the SQL console.
2. Enter the SQL statement.

   The following rules apply:
   - You can write SQL syntax elements in either upper or lower case.
   - You can add any number of spaces and line breaks.
   - To force the system to distinguish between upper/lower-case letters in database object names (such as table names), enter the name between double quotation marks: "My_Table"
   - To comment out a line, use - - (double hyphens) at the start of the line.
   - To use name completion, press the key combination CTRL + Space. This opens a list from which you can choose: Schema and table names, SQL keywords, user-defined templates.
   - Enter multiple SQL statements, separated by the configured separator character (semicolon ; by default).

3. Execute the statement in one of the following ways:
In the context menu, choose **Execute**.

Choose the **Execute** button in the toolbar.

Press **F8**.

**Note:** You can enter several statements in the SQL console and then execute them individually. To do this, simply highlight the statement and execute. If you do not highlight an individual statement, all statements are executed.

The **Result** tab appears with the statement’s results.

**Note:**

There are several options available for customizing the behavior of SQL statement execution in the SQL console. From the main menu choose **Window > Preferences > SQL**. For more information about the options, see the section on customizing the Administration Console.

**Related Links**

*SAP HANA SQL Reference*  
*User Authorization* [page 48]

Once their identity has been verified, database users can perform database operations. The confirmation that the database user is entitled to perform the operation is called authorization.

### 2.8 Customizing the Administration Console

There are many options available for customizing the Administration Console of the SAP HANA studio.

1. From the main menu, choose **Window > Preferences > Administration Console**.
2. Make the required settings.

The available settings are described in the following tables:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show only own database catalog objects</td>
<td>If you select this option, only the database objects that belong to the database user who is currently logged on are displayed.</td>
</tr>
<tr>
<td>Fetch all database catalog objects</td>
<td>By default the SAP HANA studio fetches a limited number of catalog objects when folders in the Navigator view such as <em>Tables</em> and <em>Views</em> are opened.</td>
</tr>
<tr>
<td></td>
<td>If you select this option, all catalog objects are loaded in the corresponding folder. This may affect system performance as it may take some time to fetch all database catalog objects.</td>
</tr>
<tr>
<td>Number of database catalog objects to display</td>
<td>If you do not select the <em>Fetch all database catalog objects</em> option, you can specify the maximum</td>
</tr>
</tbody>
</table>

*SAP HANA Administration Guide*  
*Working with the SAP HANA Studio*  
© 2012 SAP AG. All rights reserved.  
19
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of catalog objects to be fetched</td>
<td>number of catalog objects to be fetched. If the number of available objects exceeds the number specified here, the message <strong>Object limit reached</strong> appears. The default number is 1000.</td>
</tr>
<tr>
<td>Show table comment before table name (Modeler)</td>
<td>If you select this option, a table’s description appears before its name in the <strong>Navigator</strong> view if the Modeler perspective is active.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Confirm saving editors</td>
<td>If you select this option, the system displays a confirmation dialog box when an editor is closed with content that was not saved.</td>
</tr>
<tr>
<td>Autosaving of SQL Content</td>
<td>If you select this option, the content of console sessions is saved automatically when the SAP HANA studio is closed. No dialog requesting the user to save is displayed. Additionally, it is possible to have the content saved at a specified interval. If the SAP HANA studio is closed unexpectedly, the last version can be recovered.</td>
</tr>
<tr>
<td>Copy options</td>
<td>These are formatting options for copying content from the table editor.</td>
</tr>
<tr>
<td>Separate data with:</td>
<td></td>
</tr>
<tr>
<td>Tab separated</td>
<td></td>
</tr>
<tr>
<td>Align copied values with space</td>
<td></td>
</tr>
<tr>
<td>Copy cell in editor by using <strong>CTRL</strong></td>
<td></td>
</tr>
<tr>
<td>Copy editor content with column header</td>
<td></td>
</tr>
<tr>
<td>Representation of null value</td>
<td>This setting specifies the character used to display NULL values</td>
</tr>
<tr>
<td>Database identifier upper case</td>
<td>If you select this option, the IDs of database objects can be entered only in uppercase letters.</td>
</tr>
<tr>
<td>Default action for database tables:</td>
<td>This setting specifies which view of a table is opened when it is double-clicked in the <strong>Navigator</strong> view: its definition or its content.</td>
</tr>
<tr>
<td>Show Content</td>
<td></td>
</tr>
<tr>
<td>Show Definition</td>
<td></td>
</tr>
<tr>
<td>Table Distribution Editor</td>
<td>This setting specifies the maximum number of tables that are displayed when you show table distribution.</td>
</tr>
</tbody>
</table>
### Table 3: Global Settings

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Update state of databases on startup</td>
<td>If you select this option, the state of all registered databases is determined when the SAP HANA studio starts.</td>
</tr>
<tr>
<td>○ Confirm opening of <em>Merged Diagnosis Files</em> dialog</td>
<td>These options control the appearance of information dialogs on the <em>Diagnosis Files</em> tab.</td>
</tr>
<tr>
<td>○ Open information dialog after deleting files</td>
<td></td>
</tr>
<tr>
<td>○ Open information dialog after trace files</td>
<td></td>
</tr>
<tr>
<td>Request confirmation before a user is deleted.</td>
<td>When a user is deleted, all dependent objects are also deleted. Select this option if you want a confirmation message to appear before a user is deleted.</td>
</tr>
</tbody>
</table>

### Table 4: Plan Visualizer

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information display mode</td>
<td>You can specify the level of detail of SQL execution plans visualized using the <em>Visualize Plan</em> feature of the SQL console.</td>
</tr>
<tr>
<td>Node width and height</td>
<td>You can specify the width and height of the elements in visualized plans.</td>
</tr>
<tr>
<td>Appearance</td>
<td>You can specify how elements in visualized plans are color coded.</td>
</tr>
</tbody>
</table>

### Table 5: Result

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limit for LOB columns (Bytes)</td>
<td>This setting specifies the maximum number of bytes that are loaded from the database for one large object (LOB) column.</td>
</tr>
<tr>
<td>Limit for zoom (Bytes)</td>
<td>This setting specifies the maximum number of bytes that the SAP HANA studio displays when you zoom the LOB column in the result table in the Result editor.</td>
</tr>
<tr>
<td>Append exported data to file</td>
<td>If you select this option, then when you export the result table to a file, the system attaches the content of the current result table to the existing file content.</td>
</tr>
<tr>
<td>Display char byte value as hex</td>
<td>If you select this option, data of the data type CHAR BYTE is displayed as hexadecimal digits. If you do not select this option, this data is displayed in binary format.</td>
</tr>
<tr>
<td>Format values</td>
<td>If you select this option, country-specific formatting is applied (for example, numeric values or dates).</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Display the duration of fetching a row</td>
<td>If you select this option, you can see in the SQL console how long it took to fetch one row of a result set.</td>
</tr>
<tr>
<td>Max displayed rows in result</td>
<td>This setting specifies the maximum number of rows fetched from the database and displayed in the result table of the Result editor.</td>
</tr>
<tr>
<td>Enable zoom of LOB columns</td>
<td>You must select this option if you want to be able to zoom LOB columns in the result table of the Result editor. You can zoom an LOB column by right-clicking and choosing [Export Cell To &gt; Zoom]. Note that if you zoom an LOB column, it is automatically closed after 15 minutes or when the Result editor is closed.</td>
</tr>
</tbody>
</table>

Table 6: SQL

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stop batch SQL statement execution on error</td>
<td>If you select this option, then when you execute a list of SQL statements that are separated by comment characters, the system stops the execution when an error occurs.</td>
</tr>
<tr>
<td>Clear SQL console log before SQL statement execution</td>
<td>If you select this option, the log from the last SQL statement is deleted before the next SQL statement is executed.</td>
</tr>
<tr>
<td>Close results before SQL statement execution</td>
<td>If you select this option, then when you execute an SQL statement in the SQL console, all old results windows in that SQL console session are closed.</td>
</tr>
<tr>
<td>Display time of statement execution start</td>
<td>If you select this option, you can see in the SQL console when a statement was executed.</td>
</tr>
<tr>
<td>Display the duration of failed statements</td>
<td>If you select this option, you can see in the SQL console how long a statement took to execute in the SAP HANA studio even if the statement failed.</td>
</tr>
<tr>
<td>Connection parameters for SQL window:</td>
<td></td>
</tr>
<tr>
<td>○ Autocommit mode</td>
<td></td>
</tr>
<tr>
<td>○ Isolation level</td>
<td></td>
</tr>
<tr>
<td>○ Confirm change of connection</td>
<td></td>
</tr>
<tr>
<td>○ Auto-commit mode:</td>
<td></td>
</tr>
<tr>
<td>○ On: The system performs all COMMIT actions automatically.</td>
<td></td>
</tr>
<tr>
<td>○ Off: You have to enter COMMIT statements explicitly.</td>
<td></td>
</tr>
<tr>
<td>○ Isolation level determines how the database system implicitly controls locking and versioning of database objects.</td>
<td></td>
</tr>
<tr>
<td>○ Confirm change of connection</td>
<td></td>
</tr>
</tbody>
</table>
In the SQL console, you can change the SQL connection you are working on. When you change a connection, cursors may be closed or transactions may be rolled back. If you select this option, a change of SQL connection must first be confirmed.

Command separator
This setting specifies the separator for SQL statements in the SQL console.

Number of tables for table name completion
This setting specifies the number of tables that are displayed in the list when you use name completion in the SQL console.

### Table 7: Table Viewer

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show gridlines</td>
<td>Use these options to customize the appearance of list displays in the Administration editor, for example, the list of files on the Diagnosis Files tab.</td>
</tr>
<tr>
<td>Alternating colored rows</td>
<td></td>
</tr>
</tbody>
</table>

### Table 8: Templates
The settings available under **Templates** always refer to the editor that is currently open.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Word to be completed when you press the key combination <code>CTRL + Space</code>. You can create more than one template with the same name. If more than one template exists for one word, the system displays a list.</td>
</tr>
<tr>
<td>Context</td>
<td>Editor in which you can use the template.</td>
</tr>
<tr>
<td>Description</td>
<td>Description</td>
</tr>
<tr>
<td>Auto insert</td>
<td>If on, the code assist automatically inserts the template if it is the only proposal available at the cursor position.</td>
</tr>
</tbody>
</table>
3 Managing SAP HANA Systems in the SAP HANA Studio

3.1 Adding SAP HANA Systems in the SAP HANA Studio

To work with and manage an SAP HANA system in the SAP HANA studio, you must create a connection to the system and configure communication.

Before you can add an SAP HANA system in the SAP HANA studio, all the relevant ports in your firewall must be open. For more information, see the SAP HANA Security Guide.

1. In context menu of the Navigator view, choose Add System...
2. Enter the following information:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host name</td>
<td>Name of the host where the system is installed. If you are adding a distributed system, then you specify the host of one of its index servers. Every index server in a system has information about all the other index servers within the same system. Note: The host name of the server that hosts the database must be accessible from the client on which the SAP HANA studio is running, even if you add the system using its IP address.</td>
</tr>
<tr>
<td>Instance number</td>
<td>Instance number of the system</td>
</tr>
<tr>
<td>Use HTTPS</td>
<td>Indication of whether or not the system is to be reached through a secure connection</td>
</tr>
<tr>
<td>Description</td>
<td>The description of the system that you want to appear next to the system name in the Navigator view</td>
</tr>
<tr>
<td>Folder</td>
<td>If you are organizing your systems in the Navigator view using folders and have already created folders, choose the folder to which you want to add the system.</td>
</tr>
<tr>
<td>Locale</td>
<td>Your location This setting specifies the language of objects created in the SAP HANA repository.</td>
</tr>
</tbody>
</table>

3. Choose Next.

4. Choose the authentication type for user logon to the database:
   ○ If you are implementing single sign-on, choose Authentication by current operating system user.
   ○ If you are implementing internal authentication, choose Authentication by database user and enter the user name and password.

5. If you want to use a secure connection, choose Connect using SSL.

   Note: You must select this option to be able to modify the SSL connection properties (steps 7 and 8).

6. Choose Next.

   You can modify the following connection properties for your system:
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>Host name and SQL port through which the connection to the database is established. If you have a distributed system, the master host is entered and is used to establish the connection to the database. If this master host is unavailable, the connection is established through one of the other hosts in the system. You do not have to specify these hosts explicitly here in the connection properties of the system as they are determined automatically. Hosts that are added to the system later are also detected automatically.</td>
</tr>
<tr>
<td>Auto-Reconnect</td>
<td>If you select this option, the SAP HANA studio automatically reconnects if the connection to the system fails.</td>
</tr>
</tbody>
</table>

7. Choose **Validate SSL Certificate** to ensure that the SSL connection is secure.

To override the system host name in the certificate, enter a host name with a defined certificate.

8. Choose **Use user key store as trust store** to validate the server certificate’s validity (whether signed by a known certificate issuer) using the user key store.

The trust store property configures the trust store for the SSL connection that is used to validate that the SSL certificate of the server is issued by a trusted entity. Each user can import certificates into his or her own user key store in Java using the keytool option (part of JRE installation). The user key store is located in the home directory of the current operating system user. The filename is `.keystore`.

The set of root certificates delivered with the JRE from well-known issuers (for example, Verisign, Thawte, Deutsche Telekom) is used when this option is not selected.

9. Choose **Finish**.

The system is added in the **Navigator** view, labeled with its system ID (SID), connected user, and description. Systems that you added using a secure connection are shown with a padlock icon. When you expand a system node in the **Navigator** view, you see the default catalog, which contains public synonyms, schemas with column views (info cubes), functions, indexes, procedures, sequences, (private) synonyms, tables, and views.

**Related Links**

*SAP HANA Security Guide*

*User Authentication* [page 43]

To be able to use the SAP HANA database, the identity of the database users first needs to be verified in a process called authentication. Several types of authentication are possible.

*Disabling Default User Filtering of Schemas* [page 58]

In the **Navigator** view, the schemas available in the catalog of a system are filtered according to user by default. If, as a database administrator, you need to see all available schemas, you must disable the default schema filter.

### 3.2 Adding Users to an SAP HANA System

If you want to work with an SAP HANA system using several database users, you can create a connection to the system in the SAP HANA studio with the credentials of additional users.

Before you can connect to a system as an additional user, the following prerequisites must be met:

- The SAP HANA system has already been added once in the **Navigator** view.
• Database users exist.

1. In the Navigator view, right-click the system and choose Additional User.

2. Choose the authentication type for user logon to the system:
   ○ If you are implementing single sign-on, choose Authentication by current operating system user.
   ○ If you are implementing internal authentication, choose Authentication by database user and enter the user name and password.

3. Change the connection properties if necessary.

The system is added in the Navigator view, labeled with its system ID (SID), connected user, and description.

Related Links
SAP HANA Security Guide
User Authentication [page 43]
   To be able to use the SAP HANA database, the identity of the database users first needs to be verified in a process called authentication. Several types of authentication are possible.

User Provisioning [page 50]
As a database user with privileges for user management, you can set up other users to work with the SAP HANA database.

Disabling Default User Filtering of Schemas [page 58]
   In the Navigator view, the schemas available in the catalog of a system are filtered according to user by default. If, as a database administrator, you need to see all available schemas, you must disable the default schema filter.

3.3 Organizing SAP HANA Systems Using Folders

If you add several SAP HANA systems in the SAP HANA studio, you can organize them in the Navigator view by defining a folder structure and then assigning the systems to individual folders.

Once folders have been created, you can assign a new system to a specified folder when you add it in the Navigator view.

1. From the main menu, choose New Folder.
2. Enter a folder name.
3. In the Navigator view, move your system to the new folder using drag and drop.
4. Repeat this procedure until you have added all your systems.

3.4 Exporting List of SAP HANA Systems

You can export a list of your SAP HANA systems from the SAP HANA studio and then import them into another instance of the SAP HANA studio.

1. From the main menu, choose File Export...
2. Expand the SAP HANA Studio folder and then choose Landscape.
3. Choose Next.
4. Select the systems you want to export and enter a target file location.
5. Choose Finish.

The list of systems and their properties (name, description, hostname, instance, and so on) is exported as an XML file to the specified location.

### 3.5 Importing List of SAP HANA Systems

You can import a list of SAP systems that you previously exported from another instance of the SAP HANA studio.

1. From the main menu, choose File > Import ...
2. Expand the SAP HANA Studio folder and then choose Landscape.
3. Choose Next.
4. Choose Browse... and select the file containing the list of systems that you want to import.
5. Select the folder into which you want to import the file.

The systems appear in the Navigator view of the SAP HANA studio.

### 3.6 Searching for SAP HANA Systems

You can search the Navigator view for an SAP HANA system.

1. From the Navigator view toolbar, choose the (Find System) button.
2. Enter a search string.
   - You can also use * or ? as wildcards.
   - Matching systems are displayed.
3. Select the system you were searching for.
   - You can select several systems in the search results by pressing the key while selecting. You can use this, for example, to mark duplicate systems.
4. Choose whether you want to open the selected system in the Administration editor and/or the SQL console.

The system opens in the Administration editor and/or SQL console. If you did not select either of these options, the system is only highlighted in the Navigator view.
4 Starting and Stopping SAP HANA Systems

4.1 Starting SAP HANA Systems

To be able to start an SAP HANA system you must have the credentials of the operating system user (<sid>adm) that was created when the system was installed.

The SAP Start service (sapstartsrv) is the standard SAP mechanism for starting and stopping systems. It starts all necessary database services, such as the nameserver, indexserver, and statisticsserver services.

1. In the Navigator view, right-click the system you want to start and choose Start...
2. Optional: Specify a start timeout. The start timeout defines how long sapstartsrv waits for a service to start. If the end of the timeout period is reached, the remaining services are not started.
3. Enter the user name and password of the operating system user that was created when the system was installed (that is, <sid>adm).

The Administration editor opens in diagnosis mode and the database services start one by one. When all services have started, a green dot appears in the system icon in the Navigator view.

Related Links
Operating System User [page 42]

The <sid>adm user is not a database user but a user at the operating system level. Also referred to as the operating system administrator, this user has unlimited access to all local resources related to SAP systems.

4.2 Stopping SAP HANA Systems

To be able to stop an SAP HANA system, you must have the credentials of the operating system user (<sid>adm) that was created when the system was installed.

1. In the Navigator view, right-click the system you want to stop and choose Stop...
2. Select how you want to stop the system:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard</td>
<td>A hard shutdown forces all database services on all hosts to stop immediately.</td>
</tr>
</tbody>
</table>

*Note:* Because a hard shutdown does not force a savepoint operation, a subsequent restart may take longer.
3. Enter the user name and password of the operating system user that was created when the system was installed (that is, <sid>adm).

The Administration editor opens in diagnosis mode and the database services stop one by one. When all services have stopped, a red dot appears in the system icon in the Navigator view.

Related Links

Operating System User [page 42]

The <sid>adm user is not a database user but a user at the operating system level. Also referred to as the operating system administrator, this user has unlimited access to all local resources related to SAP systems.

4.3 Restarting SAP HANA Systems

To be able to restart an SAP HANA system you must have the credentials of the operating system user (<sid>adm) that was created when the system was installed.

In certain situations, you may have to restart the system, for example, after a power failure.

1. In the Navigator view, right-click the system you want to start and choose Restart....

2. Select how you want to stop the system:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soft</td>
<td>A soft shutdown triggers a savepoint operation before stopping all database services. During the savepoint operation, all modified data is written to disk. You can also specify a timeout after which a hard shutdown is triggered.</td>
</tr>
<tr>
<td>Stop wait timeout (sec)</td>
<td>This value specifies how long to wait for a service to stop. If the timeout expires, the remaining services are shut down anyway.</td>
</tr>
</tbody>
</table>

3. Enter the user name and password of the operating system user that was created when the system was installed (that is, <sid>adm).

The Administration editor opens in diagnosis mode. The database services first stop one by one and then restart one by one. The icon displayed for the system in the Navigator view changes as the status of the services changes.

When the system is restarted, the following activities are executed. Note that a regular ("soft") shutdown performs a savepoint operation, so there are no replay log entries to be processed in this case.

Note: Because a hard shutdown does not force a savepoint operation, a subsequent restart may take longer.
• The database receives the status of the last committed transaction.
• All the changes of committed transactions that were not written to the data area are redone.
• All write transactions that were open when the database was stopped are rolled back.
• All row tables are loaded into memory.

Important: By default, column tables are not loaded into memory until the first data request. However, in the metadata of the table, it is possible to specify that individual columns or the entire table are loaded into memory when the database is started.

• A savepoint is performed with the restored consistent state of the database.

Related Links
Operating System User [page 42]
The <sid>adm user is not a database user but a user at the operating system level. Also referred to as the operating system administrator, this user has unlimited access to all local resources related to SAP systems.

4.4 Stopping and Starting Database Services

To stop and (re)start database services, you must have the system privilege SERVICE ADMIN.

You can stop and start the individual database services (nameserver, indexserver, statisticsserver, xsengine and so on) running on hosts.

Example:

• A host in a distributed system failed and a standby host took over. However, the services of the failed host remain inactive even after the host is reachable again. In this case, you need to restart the services manually.
• After an update of SAP HANA extended application services (SAP HANA XS), the xsengine service needs to be restarted.

1. In the Administration editor on the Landscape > Services tab.
2. Right-click the service and choose the required option:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stop...</td>
<td>The service is stopped normally and then typically restarted.</td>
</tr>
<tr>
<td>Kill...</td>
<td>The service is stopped immediately and then typically restarted.</td>
</tr>
<tr>
<td>Reconfigure Service...</td>
<td>The service is reconfigured. This means that any changes made to parameters in the system’s configuration files are applied.</td>
</tr>
<tr>
<td>Start Missing Services...</td>
<td>Any inactive services are started.</td>
</tr>
</tbody>
</table>
4.5 Monitoring SAP HANA Systems During Stop and Start

The SAP HANA studio normally collects information about the system using SQL statements. However, when the system has not yet started, no SQL connection is available. Therefore, while the system is starting up or is stopped, the SAP HANA studio collects information about the database using the connection of the SAP Start (sapstartsrv) service. You can view this information in the Administration editor in diagnosis mode. In this way, you can analyze any problems that may occur during startup or while the system is stopped. You can also read diagnosis files even when the system is stopped.

The Administration editor opens automatically in diagnosis mode in the following situations:

- When you open the Administration editor for a system without an SQL connection
- When you initiate the start, stop, or restart of a system

You can manually open a system in diagnosis mode by choosing the (Open Diagnosis Mode) button from the drop-down menu of the (Administration) button in the Navigator view.

**Note:** To be able to open the Administration editor of a system in diagnosis mode, you must be able to log on using the credentials of the operating system administrator (user `<sid>`adm) that was created when the system was installed.

Related Links

*Operating System User* [page 42]

The `<sid>`adm user is not a database user but a user at the operating system level. Also referred to as the operating system administrator, this user has unlimited access to all local resources related to SAP systems.

*Monitoring Overall System Status* [page 62]

When you open the Administration editor for a particular SAP HANA system, the *Overview* tab provides you with a summary of the overall status of the system, as well as an overview of resource usage.
5 Configuring SAP HANA System Properties

5.1 Changing System Properties

The properties of an SAP HANA system are defined in the parameters of its configuration files. Configuration files are separated into sections; sections bundle parameters of the same category.

To be able to change the parameters of configuration files, you must have the system privilege INIFILE ADMIN.

1. In the Administration editor, choose the Configuration tab. A list of all configuration files appears.

2. Expand the configuration file that you want to change. All the sections of the configuration file are listed.

3. Expand the required section. All the parameters of the section are listed. For each parameter, you can see the default value.

4. In the context menu of the configuration parameter that you want to change, choose Change... The Change Configuration Value dialog box appears.

5. Enter the new value. If host-specific values are possible, you can expand the Hosts area of the Change Configuration Value dialog box, select the relevant host(s), and enter the host-specific value(s).

   It is possible to enter both a value for the system as a whole and for individual hosts. In this case, the system-specific value only applies to those hosts that do not have a host-specific value.

   If it is not possible to enter a different value for each host, the disabled icon (●) is displayed in the Host column of the list view, and there is no Hosts area in the Change Configuration Value dialog box.

After you have entered a new value for a parameter at system level, it is displayed in the System column with a green circle (●). After you have entered a new value for a parameter at host level, a gray rhombus (◇) appears in the Host column. To show information on a specific host, select the host from the Host filter.

To apply your change, you do not have to restart the system. If necessary, the system automatically restarts the relevant components.

5.2 Resetting System Properties

You can restore changed parameters in the configuration files of an SAP HANA system back to their default values.

To be able to reset the parameters of configuration files, you must have the system privilege INIFILE ADMIN.
1. In the Administration editor, choose the **Configuration** tab. A list of all configuration files appears.

2. Expand the configuration file that you want to change. All the sections of the configuration file are listed.

3. Expand the required section. All the parameters in the section are listed. You can identify parameters that have user-defined values at system level and/or host level with a green circle (●) and gray rhombus (◇) respectively.

4. To delete a user-defined value and restore the default value, you can choose one of the following methods:

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete with automatic reset:</td>
<td>The user-defined value(s) are cleared and the default value(s) are re-applied.</td>
</tr>
<tr>
<td>1. In the context menu of the</td>
<td></td>
</tr>
<tr>
<td>configuration parameter, choose</td>
<td></td>
</tr>
<tr>
<td>Delete.</td>
<td></td>
</tr>
<tr>
<td>2. Choose the layer whose user-</td>
<td></td>
</tr>
<tr>
<td>defined values you want to</td>
<td></td>
</tr>
<tr>
<td>delete.</td>
<td></td>
</tr>
<tr>
<td>3. Choose Delete.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Manually restore default:</td>
<td>The user-defined value(s) are cleared and the default value(s) are re-applied.</td>
</tr>
<tr>
<td>1. In the context menu of the</td>
<td></td>
</tr>
<tr>
<td>configuration parameter, choose</td>
<td></td>
</tr>
<tr>
<td>Change...</td>
<td></td>
</tr>
<tr>
<td>2. For the required layers,</td>
<td></td>
</tr>
<tr>
<td>choose Restore Default, or if</td>
<td></td>
</tr>
<tr>
<td>you want to reset all visible</td>
<td></td>
</tr>
<tr>
<td>layers, choose Restore Default</td>
<td></td>
</tr>
<tr>
<td>for All.</td>
<td></td>
</tr>
<tr>
<td>3. Choose Save.</td>
<td></td>
</tr>
</tbody>
</table>

### 5.3 Setting the global_allocation_limit Parameter

The SAP HANA database preallocates a pool of memory from the operating system over time, up to a predefined global allocation limit. You can change the default global allocation limit in the `global.ini` configuration file.

To be able to change the parameters of configuration files, you must have the system privilege `INIFILE ADMIN`.

The `global_allocation_limit` parameter is used to limit the amount of memory that can be used by the database. The value is the maximum allocation limit in MB. A missing entry or a value of 0 results in the system using the default settings (that is, 90% of the physical memory or physical memory minus 1GB in case of small physical memory).

1. In the Administration editor, choose the **Configuration** tab. The configuration files that contain the configuration information for the system are displayed.
2. Expand the `global.ini` configuration file and then the memorymanager section.

3. In the context menu for the `global_allocation_limit` parameter, choose `Change...` The `Change Configuration Value` dialog box appears.

4. Enter a value for the entire system and/or individual hosts.

   If you enter only a value for the system, it is used for all hosts. For example, if you have 5 hosts and set the limit to 5 GB, the database can use up to 5 GB on each host (25 GB in total). If you enter a value for a specific host, then for that host, the specific value is used and the system value is only used for all other hosts. This is relevant only for distributed systems.

**Related Links**

Allocated Memory Pools and Allocation Limits [page 97]

The SAP HANA database, across its different processes, reserves a pool of memory before actual use. This pool of allocated memory is preallocated from the operating system over time, up to a predefined `global allocation limit`, and is then efficiently used as needed by the SAP HANA database code.

### 5.4 Changing the Default SLD Data Supplier Configuration

The System Landscape Directory (SLD) is the central directory of system landscape information relevant for the management of your software lifecycle. Data suppliers collect and send system data to SLD on a regular basis. The name server is the SLD system for SAP HANA systems.

Before you can change the default configuration of the SLD supplier, the following prerequisites must be fulfilled:

- The SLD must be configured.
- You must have the system privilege INIFILE ADMIN.

For SAP HANA systems, the name server is the SLD data supplier. It is configured by default to automatically transfer data to the SLD in XML format using the `sldreg` executable on a regular basis. However, if it is necessary to change the default settings, you can do so in the SAP HANA studio by modifying the `nameserver.ini` file. For example, it may not be necessary to send data to the SLD frequently if your landscape is stable, or you may need to change the default save locations of the configuration and log files.

1. In the Administration editor, choose the **Configuration** tab.
2. Right-click the `nameserver.ini` file and choose **Add Section**.
3. Create a new section `sld`.
4. Add those parameters whose default value you want to change.

The following table lists the possible parameters and their default values.

<table>
<thead>
<tr>
<th>Key</th>
<th>Meaning</th>
<th>Default Value</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>enable</td>
<td>Activates or deactivates the SLD data supplier</td>
<td>true</td>
<td>Allowed values are true, false</td>
</tr>
</tbody>
</table>

**Note:** Under normal circumstances, you will not need to change the default values. It should only be necessary, for example, for testing purposes or if requested as part of a support inquiry.
<table>
<thead>
<tr>
<th>Key</th>
<th>Meaning</th>
<th>Default Value</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interval</td>
<td>Specifies the frequency (in seconds) with which the CIM.xml file is generated. If a newly-generated document is the same as the previous one, it is not sent to the SLD.</td>
<td>300</td>
<td>It does not make sense to enter small positive values or negative values. If you enter 0 or a negative value, data is transferred to the SLD only once. Enter a value without a &quot;1000 separator&quot; (for example, 1.899, not 1.899 or 1.899), otherwise it will be interpreted as 0.</td>
</tr>
<tr>
<td>force_interval</td>
<td>Specifies how often (in seconds) the CIM.xml file must be sent to the SLD, even if the file has not changed.</td>
<td>43200</td>
<td></td>
</tr>
<tr>
<td>config_path</td>
<td>Specifies the location of the folder that contains the configuration file slldest.cfg. This file is a parameter for the call to sldreg.</td>
<td>/usr/sap/&lt;sid&gt;/SYS/global</td>
<td>Example: /usr/sap/MPW/SYS/global</td>
</tr>
<tr>
<td>xmlpath</td>
<td>Specifies where the file sldreg.xml is generated and where the smlreg.log log file is written. smlreg.log is the log file of sldreg, and both files are parameters for the call to sldreg.</td>
<td>/usr/sap/&lt;sid&gt;/&gt;HDB&lt;id&gt;/&lt;currenthost&gt;</td>
<td>Example: /usr/sap/LR/HDB42/velber1cm1</td>
</tr>
</tbody>
</table>

**Note:** If errors occur in the transfer of data to the SLD, you can check the log file smlreg.log and the nameserver trace (trace topics SLDCollect and SLDSend). The SLD data supplier is traced in the nameserver’s regular trace file as part of database tracing.

**Related Links**

*Other Database Traces* [page 225]

Other database traces can be written to obtain detailed information about the actions of the system, including component-specific database traces, traces for end-to-end analysis, and expensive statement traces.

**5.4.1 System Landscape Directory – Additional Resources**

More information about the System Landscape Directory

<table>
<thead>
<tr>
<th>Topic</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLD Configuration</td>
<td>● <a href="#">SAP Note 1018839</a> (Registering in the System Landscape Directory Using SLGREG)</td>
</tr>
</tbody>
</table>
5.5 Configuring Trace File Rotation

Trace file rotation prevents trace files from growing indefinitely by limiting the size and number of trace files. You can configure trace file rotation globally for all services in the system and for individual services.

To be able to configure trace file rotation, you must have the system privilege INIFILE ADMIN.

1. In the Administration Editor, choose the Configuration tab.
2. Depending on whether you are configuring trace file rotation for all system services or for an individual service, proceed as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| All services | 1. Navigate to the global.ini file and expand the section trace.  
               2. Configure the maxfiles parameter by specifying the maximum number of trace files that may exist.  
               3. Configure the maxfilesize parameter by specifying in bytes the maximum size an individual trace file may reach.                                                                 |
|              | **Note:** The default configuration for trace file rotation in the global.ini file is maxfiles=10 and maxfilesize=10000000.                                                                                       |
| Individual service | 1. Navigate to the configuration file of the relevant service (for example, indexserver.ini) and expand the section trace.  
                             If there is no trace section, create one by right-clicking the file and choosing Add Section.  
                             2. Configure the maxfiles parameter by specifying the maximum number of trace files that may exist.  
                             3. Configure the maxfilesize parameter specifying the maximum size an individual trace file may reach in bytes.                                                                 |
|              | **Note:** If the two parameters do not exist in the trace section or if you created a new trace section, create the parameters by right-clicking the section and choosing Add Parameter. |

When a trace file reaches the specified maximum file size, it is closed, and a new file created. When the specified maximum number of files is reached, the next time a new file is created, the first file is deleted, and so on.
Note: The statistics server checks the size and number of diagnosis files regularly. The threshold values for these checks (check 50 and 51) should be in line with the configured trace file rotation.

Related Links

Configuring Traces [page 221]
You can activate and configure several traces on the Trace Configuration tab of the Administration editor. Trace data is saved to trace files, which you can view on the Diagnosis Files tab.

Statistics Server Checks [page 75]
The statistics server performs regular checks and issues an alert when an alert condition is fulfilled.

Configuring Check Thresholds [page 83]
For some checks performed by the statistics server, you can configure when an alert is issued, that is the alert condition. A check can have a low, medium, and high priority threshold.
6 Managing SAP HANA Licenses

6.1 License Types

License keys are required to use SAP HANA databases. The SAP HANA database supports two kinds of license keys: temporary license keys and permanent license keys.

While temporary license keys are automatically installed in an SAP HANA database, permanent license keys have to be requested on the SAP Service Marketplace and applied to the individual SAP HANA database.

Temporary License Keys

A temporary license key, which is valid for 90 days, is automatically installed with a new SAP HANA database. During this period, you should request and apply a permanent license key.

Permanent License Keys

You can request a permanent license key on the SAP Service Marketplace at https://service.sap.com/support under Keys & Requests. Permanent license keys are valid until the predefined expiration date. Furthermore, they specify the amount of memory licensed to the target SAP HANA installation. Before a permanent license key expires, you should request and apply a new permanent license key. If a permanent license key expires, a temporary license key valid for 28 days is automatically installed. During this time, you can request and install a new permanent license key.

There are two types of permanent license key available for SAP HANA: unenforced and enforced. If an unenforced license key is installed, the operation of SAP HANA is not affected if its memory consumption exceeds the licensed amount of memory. However, if an enforced license is installed, the system is locked down when the current memory consumption of SAP HANA exceeds the licensed amount of memory plus some tolerance. If this happens, either SAP HANA needs to be restarted, or a new license key that covers the amount of memory in use needs to be installed.

The two types of permanent license key differ from each other in the following line in the license key file:

<table>
<thead>
<tr>
<th>License Key Type</th>
<th>License Key File Entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unenforced</td>
<td>SWPRODUCTNAME=SAP-HANA</td>
</tr>
<tr>
<td>Enforced</td>
<td>SWPRODUCTNAME=SAP-HANA-ENF</td>
</tr>
</tbody>
</table>

Note: Although enforced license keys currently only apply to SAP Business One, it is technically possible to install such a license in an SAP HANA instance with a regular, unenforced permanent license. In this case, the unenforced license key has priority. That is, if a valid unenforced license key is found, no memory
consumption check is enforced. However, if one license key expires and becomes invalid, the other one, if valid, becomes the valid license key of the instance. If the latter is an enforced license key, then the memory consumption check is enforced.

System Lockdown

The system goes into lockdown mode in the following situations:

- The temporary license key has expired.
- You were using a temporary license key and the hardware key has changed.
- The permanent license key has expired and you did not renew it within 28 days.
- The installed license key is an enforced license key and the current memory consumption exceeds the licensed amount plus the tolerance.
- You deleted all license keys installed in your database.
- The system ID and/or hardware key of your database have changed, for example, after system copy or renaming.

In lockdown mode, no queries are possible. Only a user with the system privilege LICENSE ADMIN can connect to the database and execute license-related queries, such as, obtain previous license data, install a new license key, and delete installed license keys.

In addition, the database cannot be backed up in lockdown mode.

**Note:** If a system has locked down due to an invalid or expired license, the icon indicating the operational status of the system in the Navigator view and the System Monitor changes accordingly.

6.2 Checking the Current License Key

You can check the properties of your SAP HANA license in the SAP HANA studio.

To be able to check the current license key, you must have the system privilege LICENSE ADMIN.

1. In the Navigator view, right-click the system and choose Properties.
2. Choose License.

The Current License Key screen area appears with the following information:

- License type
- Start date of the license key
- Expiration date of the license key
6.3 Installing Permanent Licenses

To the SAP HANA database, you must request and install a valid permanent license key.

To be able to install a permanent license key, you must have the system privilege LICENSE ADMIN.

1. Get the information required to request a permanent license key.
   To request the first permanent license key for a newly installed SAP HANA database, you need to provide the hardware key and the system ID. To request a subsequent permanent license key, you need the installation number and system number of your SAP HANA database. You can get the required information in the SAP HANA studio as follows:
   a) In the Navigator view, right-click the system and choose Properties.
   b) Choose License.

   If the database is currently running on a temporary license key, the Request License Key screen area displays the hardware key and the system ID. If the database already has a valid permanent license key, the installation number and system number are displayed. Alternatively, you can use SQL to access the required information from the M_LICENSE system view.

2. Request a license key on SAP Service Marketplace under Keys & Requests Request license keys.
   When completing the request form, if you have the installation number and system number, then enter them first so that the other input fields are auto-completed. When you have finished, choose Submit.
   Permanent licenses are sent by e-mail attachment.

3. To install the license key, you have the following options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Steps</th>
</tr>
</thead>
</table>
| SAP HANA Studio | 1. In the Navigator view, right-click the system and choose Properties License.  
 |              | 2. In the Request License Key area, choose Install License Key and select the file that you received by e-mail. |
| SQL console  | Execute the SQL command SET SYSTEM LICENSE <license file content> |

1. **Note**: If you are installing a second or subsequent permanent license key, it must have the same system-identification data as the permanent license key previously installed in the database. In particular, the system ID, hardware key, installation number, and system number must be the same. If any difference is detected in this data, the installation of the license key fails and no change is made to the license key in the database.

Related Links

*Opening the SQL Console*[page 18]

Some tasks may require you to work with SQL statements, for example, certain administration tasks can only be performed using SQL. You can enter, execute, and analyze SQL statements in the SQL console.
6.4 Deleting Existing Permanent License Keys

You can delete all existing license keys in an SAP HANA database, for example, if permanent license keys with an incorrect installation number or incorrect system number were installed on the database.

To be able to delete a permanent license key, you must have the system privilege LICENSE ADMIN.

1. In the Navigator view, right-click the system and choose Properties.
2. Choose License.
3. Choose Delete License Key.

All permanent license keys are deleted. This results in the lockdown of the database. The installation of a new, valid permanent license key is required to unlock the database.

Note: You can also delete all installed license keys by executing the following SQL command

`UNSET SYSTEM LICENSE ALL`.

Related Links

Opening the SQL Console [page 18]

Some tasks may require you to work with SQL statements, for example, certain administration tasks can only be performed using SQL. You can enter, execute, and analyze SQL statements in the SQL console.
7 Managing SAP HANA Users

7.1 Database Users

Every user who wants to work with the SAP HANA database must have a database user with the required privileges.

The right to access resources and to perform operations in SAP HANA is determined exclusively by the privileges of the database user who attempts to perform these operations. SAP HANA supports the concept of a role, which is fundamentally a set of privileges. Roles are granted to database users, and users thereby gain the respective privileges.

Standard Users

When you install the SAP HANA database, a database user, called SYSTEM, is created by default. The database user SYSTEM has irrevocable system privileges, such as the ability to create other database users, access system tables, and so on.

Note: For security reasons, it is highly recommended that you do not use the SYSTEM database user for day-to-day activities. Use SYSTEM to create database users with the minimum privilege set required for their duties, and use those users for day-to-day administrative activities.

Several technical database users (that is, database users that do not correspond to real people) are also created, for example, SYS and _SYS_STATISTICS. These users cannot log on to the SAP HANA database.

Related Links
SAP HANA Security Guide
SAP HANA System Tables and Monitoring Views Reference

7.2 Operating System User

The <sid>adm user is not a database user but a user at the operating system level. Also referred to as the operating system administrator, this user has unlimited access to all local resources related to SAP systems.

In addition to the SAP HANA database user SYSTEM, the installation process also creates an external operating system user (<sid>adm, for example, sp1adm or xyzadm). This operating system user, also referred to as the operating system administrator, simply exists to provide an operating system context. From the operating system perspective, the operating system administrator is the user that owns all SAP HANA files and all related operating system processes. Within the SAP HANA studio, the operating system user’s credentials are required, for example, to start or stop database processes or to execute a recovery.

The operating system user is not an SAP HANA database user.
7.3 User Authentication

To be able to use the SAP HANA database, the identity of the database users first needs to be verified in a process called authentication. Several types of authentication are possible.

Internal Authentication

Users are created in SAP HANA database only. Authentication is handled by the SAP HANA database by means of a username-password combination.

Authentication Using External User Repositories

The SAP HANA database supports the mapping of users created in the SAP HANA database to external users in the following ways:

- Integration of the Kerberos network authentication protocol
  
  If you want users to be able to log on to the SAP HANA database using Kerberos, you must install the relevant Kerberos client software from your operating system software package and configure the SAP HANA database accordingly. The users stored in the Kerberos Key Distribution Center can then be mapped to database users in SAP HANA database.

- Security Assertion Markup Language (SAML)
  
  To be able to use SAML authentication, at least one SAML identity provider must have been configured.

Related Links

SAP HANA Security Guide

7.3.1 Working with the Password Blacklist

When creating new passwords, users are not allowed to use blacklisted words or partial words. Use SQL commands to insert or delete words or partial words from the password blacklist table.

- For inserting terms into and deleting terms from the password blacklist, you must have INSERT and DELETE privileges on either the table `_SYS_PASSWORD_BLACKLIST` or the entire `_SYS_SECURITY` schema.
- To view the contents of a table only, you must have the SELECT privilege.

The password blacklist in SAP HANA has been implemented with the table `_SYS_PASSWORD_BLACKLIST`. This table is empty when you create a new instance.

You can add records to and delete records from the `_SYS_PASSWORD_BLACKLIST` table using the INSERT and DELETE SQL commands. You must specify values for all three columns described below:
### Column Name Description

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLACKLIST_TERM</td>
<td>The word or partial word you want to insert into or delete from the _SYS_PASSWORD_BLACKLIST table.</td>
</tr>
<tr>
<td>CHECK_PARTIAL_PASSWORD</td>
<td>• TRUE&lt;br&gt;For partial words or terms&lt;br&gt;• FALSE&lt;br&gt;For whole words or terms</td>
</tr>
<tr>
<td>CHECK_CASE_SENSITIVE</td>
<td>• TRUE&lt;br&gt;For case-sensitive words or partial words&lt;br&gt;• FALSE&lt;br&gt;For non-case-sensitive words or partial words</td>
</tr>
</tbody>
</table>

**INSERT INTO _SYS_SECURITY._SYS_PASSWORD_BLACKLIST VALUES ('sap', 'TRUE', 'FALSE')**

In this example, the passwords "SAP", "my_sap_pwd" and "sap_password" would not be allowed, regardless of how the password layout and minimal password length are defined in the corresponding parameters.

### 7.3.2 Configuring Kerberos for SAP HANA Database Hosts

SAP HANA supports Kerberos version 5 for single sign-on based on Active Directory (Microsoft Windows Server) or Kerberos authentication servers. Both ODBC database clients and JDBC database clients support the Kerberos protocol.

To allow users to log on to the SAP HANA database using Kerberos authentication, you must install MIT Kerberos client libraries on the host(s) of the SAP HANA database.

Once Kerberos client libraries have been installed, you must configure Kerberos on the authentication server by performing the following logical steps:

1. Register one service principal name (SPN) for each host in the SAP HANA system using the following syntax:
   
   hdb/<host domain name>@<Kerberos realm name>, where
   
   ○ <host domain name> is the fully qualified domain name of the host
   ○ <Kerberos realm name> (Kerberos terminology) is identical to “domain name” in Active Directory terminology
   
   This results in the generation of a service key table (keytab) for each host. This keytab contains the encrypted key for the host in question.

2. Export the keytab(s) to files.

3. Import each keytab file into the Kerberos installation on the respective host.

The concrete steps to be performed on the authentication server depend on whether you are using Kerberos or Active Directory as follows:
1. Register the SPN.

Note: In Active Directory, before a SPN can be registered, you must create a plain user account that acts as the server principal on the domain controller. Afterward, you must map the SPN to the user account using a separate command.

2. Export the keytab(s) to files using a command line tool shipped with the authentication server. This is applicable for both Kerberos and Active Directory.

3. Import the keytab files. The files are transported to the file system path on the SAP HANA database hosts in line with how the Kerberos client is configured.

You can now map the users stored in the Kerberos Key Distribution Center to database users in SAP HANA database. You can do this when you create database users. Alternatively, if database users already exist, you can change their authentication details.

For more information, see Kerberos or Microsoft Windows Server documentation.

Related Links

Creating Users [page 53]
You create a database user for every user who wants to work with the SAP HANA database. You configure how the user will be authenticated, as well as grant the privileges and roles necessary for the user’s function.

Changing Users [page 54]
You can change a user’s authentication information, grant them new privileges and roles, as well as revoke previously granted privileges and roles.

SAP HANA Security Guide

7.3.3 Configuring SAML Identity Providers
If you are implementing Security Assertion Markup Language (SAML) user authentication, you must configure SAML identity providers for the required users.

If you are implementing Security Assertion Markup Language (SAML) user authentication, you must configure SAML identity providers for the required users.

Recommendation: To avoid replay attacks, we recommend that you set up secure communication between the individual components of the SAP HANA database and client connections using the secure sockets layer (SSL) protocol when implementing SAML authentication.

When you configure an SAML identity provider, you specify the X.509 certificate that will be used to sign the SAML assertions from the identity provider. This means that you must have already imported the relevant certificates into the trust store of your SAP HANA instance using command line tools, in line with the configured security provider (OpenSSL or SAP Crypto). You must ensure that the entire certificate chain of the X.509 certificate is in the trust store.

You can then configure the SAML identity provider as follows:

1. In the Navigator view, right-click the system and choose Properties ➤ Security ➤ SAML Configuration. The SAML Configuration dialog box appears.

2. Choose Add.
3. Enter the name of the identity provider.
   The following naming conventions apply:
   ○ Spaces and special characters except underscore (_) are not permitted.
   ○ The name must start with a letter.
   ○ The name cannot exceed 127 characters.

4. Choose Read Certificate.
   The system reads the X.509 certificate obtained from the identity provider and extracts the issuer and subject distinguished names (DNs). It then enter these in the corresponding fields.

   **Note:** If the certificate fails to read with an IOExeption or a CertificateException, try recoding the certificate from Base64 (*.pem) to DER (*.der) using OpenSSL or other tools.
   You can also enter the issuer and subject DNs manually.

5. Choose OK.
   The identity provider is now available for mapping to a database user. You can do this when you create the database user. Alternatively, if the database user already exists, you can change their authentication details.

   Related Links
   - Creating Users [page 53]
     You create a database user for every user who wants to work with the SAP HANA database. You configure how the user will be authenticated, as well as grant the privileges and roles necessary for the user’s function.
   - Changing Users [page 54]
     You can change a user’s authentication information, grant them new privileges and roles, as well as revoke previously granted privileges and roles.

   **SAP HANA Security Guide**

   **7.3.4 Problems with User Authentication**
   Authentication problems manifest themselves as failed user login. In many cases, the reason for the failure will not be clear to the user. You must analyze the database trace to determine the cause of the problem.

   For security reasons, no information about error conditions are provided to a user directly after a failed logon attempt, since this could be abused by attackers. In case of authentication problems, the affected user must contact the system administrator, who will then analyze the database trace on the server side. Relevant information can be found in the SAP HANA database trace for the index server component. You can activate the database trace on the Trace Configuration tab of the Administration editor. The trace component is authentication.

   **Kerberos**
   Kerberos authentication is implemented in the SAP HANA database using the Generic Security Services Application Program Interface (GSS API). Since this is an internet standard (cf. RFC 4121), all Kerberos-related errors are traced under the authentication component in the following generic way:

   `<SAP HANA DB error text> (<GSS major code>,<GSS minor code> - <GSS major text> <GSS minor text>)`

   GSS API error texts are sometimes difficult to relate to the concrete problem. In the following, some hints for selected trace messages are given.
GSS API Error Code | Error Text | Comment
---|---|---
851968.2529639142 | Minor error text: "Key version number for principal in key table is incorrect" | Hint: The service key table (keytab) in use on the SAP HANA database host does not match the one created on authentication server.
| |  | Solution: Re-export the keytab file from the authentication server and re-import it into the host’s Kerberos installation.

851968.39756033 | SAP HANA database error text: "Cannot get keytab entry for host: <FQDN>"  
Minor error text: "No principal in keytab matches desired name" | Hint: Keytab actually used might be different than expected (default: /etc/krb5.keytab).
| |  | Solution: Check environment variable KRB5_KTNAME.

851968.2529639136 | HANA DB error text: "Cannot get keytab entry for host: <FQDN>"  
Minor error text: "Configuration file does not specify default realm" | Hint: Kerberos configuration file actually used might be different than expected (default: /etc/krb5.conf).
| |  | Solution: Check environment variable KRB5_CONFIG.

There are many potential problems setting up a Kerberos infrastructure that are not related to the SAP HANA system in particular but relevant for any Kerberos-based authentication. For further information, refer to the documentation provided with MIT Kerberos or Microsoft Server/Active Directory.

## SAML

<table>
<thead>
<tr>
<th>Error Situation</th>
<th>Cause</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>User cannot connect to the database with SAML assertion.</td>
<td>The issuer and subject distinguished names (DNs) in the SAML assertion do not match those configured in the identity provider.</td>
<td>Investigate which issuer and subject DNs were used in the SAML assertion. You will find them in the trace file indexserver_alert_&lt;hostname&gt;.trc. Compare these with those configured in the service provider.</td>
</tr>
</tbody>
</table>

Related Links

* Configuring Kerberos for SAP HANA Database Hosts [page 44]*

SAP HANA supports Kerberos version 5 for single sign-on based on Active Directory (Microsoft Windows Server) or Kerberos authentication servers. Both ODBC database clients and JDBC database clients support the Kerberos protocol.
Configuring SAML Identity Providers [page 45]
If you are implementing Security Assertion Markup Language (SAML) user authentication, you must configure SAML identity providers for the required users.

Diagnosis Files [page 219]
Diagnosis files include log and trace files, as well as a mixture of other diagnosis, error, and information files. In the event of problems with the SAP HANA database, you can check these diagnosis files for errors.

7.4 User Authorization

Once their identity has been verified, database users can perform database operations. The confirmation that the database user is entitled to perform the operation is called authorization.

The user must have both the privilege to perform the operation and access rights to the resources (such as schemas and tables) to which the operation applies.

Privileges can be granted to database users either directly, or indirectly through roles that they have been granted. In this case, the privileges are inherited. Roles are the standard mechanism of granting privileges to users.

The following privilege types exist in the SAP HANA database:

<table>
<thead>
<tr>
<th>Privilege Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System privilege</td>
<td>System privileges are SQL privileges that control general system activities and are mainly for administrative purposes, such as creating schemas, creating and changing users and roles, performing data backups, managing licenses, and so on.</td>
</tr>
<tr>
<td>Object privilege</td>
<td>Object privileges are SQL privileges that are used to restrict access to and modification of database objects, such as tables and views. Depending on the object type, different actions can be authorized (for example, SELECT, CREATE ANY, ALTER, DROP, and so on).</td>
</tr>
<tr>
<td>Analytic privilege</td>
<td>Analytic privileges are used to restrict read access to data in SAP HANA information models (that is analytic views, attribute views, and calculation views) depending on certain values or combinations of values. Analytic privileges are evaluated during query processing.</td>
</tr>
<tr>
<td>Package privileges</td>
<td>Package privileges are used to restrict access to and the ability to work in packages in the repository of the SAP HANA database. Packages contain design time versions of various objects, such as analytic views, attribute views, calculation views, and analytic privileges.</td>
</tr>
</tbody>
</table>

Note: An additional privilege type is supported for SAP HANA Extended Application Services (XS) applications. Application developers can create application privileges to authorize user and client access to their application. Application privileges can be granted to users through procedures. To grant or revoke the privilege to or from a user directly, you have to use the GRANT_APPLICATION_PRIVILEGE or REVOKE_APPLICATION_PRIVILEGE procedure in the _SYS_REPO schema. It is not possible to grant application privileges to users in the SAP HANA studio.

Related Links
### 7.4.1 Privileges for Administration Tasks

To perform administration tasks, you require certain privileges.

The following table provides you with an overview of the privileges required for specific administration tasks.

<table>
<thead>
<tr>
<th>Task</th>
<th>Privilege</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open the Administration Console with read-only access to the system, monitoring views, and statistics server tables</td>
<td>System privilege CATALOG READ or DATA ADMIN and object privilege SELECT for schema _SYS_STATISTICS</td>
</tr>
<tr>
<td>Stop and start database services</td>
<td>System privilege SERVICE ADMIN</td>
</tr>
<tr>
<td>Install and delete license keys, check license information</td>
<td>System privilege LICENSE ADMIN</td>
</tr>
<tr>
<td>Create, change and delete roles in runtime</td>
<td>System privilege ROLE ADMIN and the privileges required to grant the necessary privileges and roles</td>
</tr>
<tr>
<td>Create, change, and delete users</td>
<td>System privilege USER ADMIN and the privileges required to grant the necessary privileges and roles</td>
</tr>
<tr>
<td>View alert information</td>
<td>Object privilege SELECT for the schema _SYS_STATISTICS</td>
</tr>
<tr>
<td>Cancel operations</td>
<td>System privilege SERVICE ADMIN</td>
</tr>
<tr>
<td>Execute table redistribution operations</td>
<td>System privilege RESOURCE ADMIN and at least the object privilege ALTER for all schemas involved</td>
</tr>
<tr>
<td>Manually move tables and table partitions to another host in a distributed system</td>
<td>System privilege DATA ADMIN or system privilege CATALOG READ and object privilege ALTER for the table being moved</td>
</tr>
<tr>
<td>Perform delta merge operation on a table</td>
<td>Object privilege UPDATE for the table</td>
</tr>
<tr>
<td>Compress a table</td>
<td>Object privilege UPDATE for the table</td>
</tr>
<tr>
<td>Import catalog objects</td>
<td>System privileges IMPORT and object privilege INSERT for the catalog objects in question</td>
</tr>
<tr>
<td>Export catalog objects</td>
<td>System privileges EXPORT and object privilege SELECT for the catalog objects in question</td>
</tr>
<tr>
<td>Configure traces (except kernel profiler)</td>
<td>System privilege TRACE ADMIN</td>
</tr>
</tbody>
</table>

**Note:** A read-only role, MONITORING, is shipped with the SAP HANA database installation. It includes the above-mentioned privileges and can be assigned to users to give them read-only access.
### Related Links

**SAP HANA SQL Reference**

Prerequisites for Granting and Revoking Privileges and Roles [page 57]

To be able to grant and revoke privileges and roles to and from users and roles, several prerequisites must be met.

### 7.5 User Provisioning

As a database user with privileges for user management, you can set up other users to work with the SAP HANA database.

The recommended process is as follows:

1. **Define and create roles.**

   ![Note](image)
   
   It is recommended that you create roles in the repository of the SAP HANA database. This approach offers more flexibility than creating them as runtime objects.

2. **Create users.**

3. **Grant roles to users.**

   ![Note](image)
   
   If you are using an Identity Management (IDM) system for user provisioning, it is highly recommended that you create a dedicated technical user for that system that has the USER ADMIN and ROLE ADMIN privileges. This database user should then be used exclusively by the IDM system for its user provisioning tasks.

### 7.5.1 Roles

A role is a collection of privileges that can be granted to either a user or another role in runtime.
A role typically contains the privileges required for a particular function or task, for example:

- Business end users reading reports using client tools such as Microsoft Excel
- Modelers creating models and reports in the modeler of the SAP HANA studio
- Database administrators operating and maintaining the database and users in the Administration editor of the SAP HANA studio

Privileges can be granted directly to users of the SAP HANA database. However, roles are the standard mechanism of granting privileges as they allow you to implement complex, reusable hierarchies of user access that can be modeled on business roles. Several standard roles are delivered with the SAP HANA database (for example, MODELING, MONITORING). You can use these as templates for creating your own roles.

Roles in the SAP HANA database can exist as runtime objects only, or as design-time objects that become runtime objects on activation.

**Role Structure**

A role can contain any number of the following privileges:

- System privileges for administrative tasks (for example, AUDIT ADMIN, BACKUP ADMIN, CATALOG READ)
- Object privileges on database objects (for example, SELECT, INSERT, UPDATE)
- Package privileges on repository packages (for example, REPO.READ, REPO.EDIT_NATIVE_OBJECTS, REPO.ACTIVATE_NATIVE_OBJECTS)
- Analytic privileges on SAP HANA information models
- Application privileges for enabling access to SAP HANA XS applications

\[\text{Note: Application privileges cannot be granted to roles in the SAP HANA studio.}\]

A role can also extend other roles.

**Role Modeling**

You can model roles in the following ways:

- As runtime objects on the basis of SQL statements
- As design-time objects in the repository of the SAP HANA database

It is recommended that you model roles as design-time objects for the following reasons.

Firstly, unlike roles created in runtime, roles created as design-time objects can be transported between systems. This is important for application development as it means that developers can model roles as part of their application's security concept and then ship these roles or role templates with the application. Being able to transport roles is also advantageous for modelers implementing complex access control on analytic content. They can model roles in a test system and then transport them into a productive system. This avoids unnecessary duplication of effort.

Secondly, roles created as design-time objects are not directly associated with a database user. They are created by the technical user _SYS_REPO and granted through the execution of stored procedures. Any user with access to these procedures can grant and revoke a role. Roles created in runtime are granted directly by the database.
user and can only be revoked by the same user. Additionally, if the database user is deleted, all roles that he or she granted are revoked. As database users correspond to real people, this could impact the implementation of your authorization concept, for example, if an employee leaves the organization or is on vacation.

Caution: The design-time version of a role in the repository and its activated runtime version should always contain the same privileges. In particular, additional privileges should not be granted to the activated runtime version of a role created in the repository. Although there is no mechanism of preventing a user from doing this, the next time the role is activated in the repository, any changes made to the role in runtime will be reverted. It is therefore important that the activated runtime version of a role is not changed in runtime.

### 7.5.2 Creating Roles in Runtime

You can create a new role directly in runtime and grant it the privileges and roles necessary for the task or function that it represents.

To be able to create a role as a runtime object, you must have the system privilege ROLE ADMIN. You must also have the privileges required to grant privileges and roles to the new role.

Note: Creating roles in the repository of the SAP HANA database offers more flexibility than creating them in runtime as described here. The recommended approach is therefore to create roles as repository objects. For more information about roles as repository and how to model roles in design time, see SAP HANA Developer Guide.

1. In the Navigator view, choose Security > Roles.
2. From the context menu, choose New Role.
3. Specify a unique role name. The role name can contain all characters, except double quotation marks ("...").
4. Assign the required roles and privileges (object privileges, analytic privileges, system privileges, and package privileges) by choosing the button and searching for the role or privilege. To allow a user assigned the role to grant assigned roles or privileges to other users and roles, select Grantable to other users and roles.

Note: This option is not available if you are granting a role that was created in the repository.

5. Choose the Deploy button to create the role.

The role is created and appears in the Security > Roles folder.

Note: You can delete a role by right-clicking it in the Navigator view and choosing Delete.

Related Links

Prerequisites for Granting and Revoking Privileges and Roles [page 57]

To be able to grant and revoke privileges and roles to and from users and roles, several prerequisites must be met.

SAP HANA Developer Guide
7.5.3 Changing Roles in Runtime
You can grant new privileges and roles to a role directly in runtime, as well as revoke previously granted privileges and roles.

To be able to change a role in runtime, you must have the privileges required to grant and revoke privileges and roles to and from the role.

Note: It is important that you do not change the activated runtime version of a role created in the SAP HANA repository as described here to avoid discrepancies between the design-time version of the role and its activated runtime version. Although you can change an activated role in runtime (for example, you can grant additional privileges to a role), the next time the role is activated in the repository, any changes made to the role in runtime will be lost. For more information about design-time roles, see SAP HANA Developer Guide.

1. In the Navigator view, choose Security > Roles.
2. Open the relevant role and make the required changes to the privileges.
3. Choose the (Deploy) button to save the role.

Related Links
SAP HANA Developer Guide

7.5.4 Creating Users
You create a database user for every user who wants to work with the SAP HANA database. You configure how the user will be authenticated, as well as grant the privileges and roles necessary for the user’s function.

To be able to create a user, you must have the system privilege USER ADMIN. You must also have the privileges required to grant privileges and roles to the new user.

If you are implementing external authentication, the following prerequisites apply:

- If you are implementing Kerberos user authentication, you have installed the relevant Kerberos client software from your operating system software package and configured the SAP HANA database accordingly.
- If you are implementing Security Assertion Markup Language (SAML) user authentication, you have created the necessary identity providers.

1. In the Navigator view, choose Security > Users.
2. From the context menu, choose New User.
3. Specify the following information:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User name</td>
<td>You must give the user a unique name. The user name can contain only letters (Aa-Zz), numbers (0-9), and underscores (_).</td>
</tr>
<tr>
<td>Authentication</td>
<td>You can set up one or more of the following types of user authentication:</td>
</tr>
<tr>
<td></td>
<td>○ Internal authentication by specifying a user name and password</td>
</tr>
<tr>
<td></td>
<td>○ Kerberos authentication (external) by specifying the user principal name (UPN) specified in the Microsoft Active Directory or the Kerberos Key Distribution Center as the external ID</td>
</tr>
</tbody>
</table>
### Option Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>○ SAML authentication (external) by selecting the identity provider and then entering the user ID known by the SAML identity provider. Alternatively, you can allow the identity provider to map users to the database user by selecting the checkbox in the Any column.</td>
<td></td>
</tr>
</tbody>
</table>

### Roles and privileges

You can grant roles and privileges (object privileges, analytic privileges, system privileges, and package privileges) to the user.

To assign privileges, choose the button.

To allow the user to pass on his or her privileges to other users, select **Grantable to other users and roles**.

**Note:** This option is not available if you are granting either a role that was created in the repository or a privilege on a repository object.

---

4. Choose the (Deploy) button to create the user.

The user’s password (internal authentication) is verified to ensure that it complies with the system’s password policy.

The user is created and appears in the **Users** folder.

### Related Links

- **Prerequisites for Granting and Revoking Privileges and Roles** [page 57]
  
  To be able to grant and revoke privileges and roles to and from users and roles, several prerequisites must be met.

- **Configuring Kerberos for SAP HANA Database Hosts** [page 44]
  
  SAP HANA supports Kerberos version 5 for single sign-on based on Active Directory (Microsoft Windows Server) or Kerberos authentication servers. Both ODBC database clients and JDBC database clients support the Kerberos protocol.

- **Configuring SAML Identity Providers** [page 45]
  
  If you are implementing Security Assertion Markup Language (SAML) user authentication, you must configure SAML identity providers for the required users.

- **Working with the Password Blacklist** [page 43]
  
  When creating new passwords, users are not allowed to use blacklisted words or partial words. Use SQL commands to insert or delete words or partial words from the password blacklist table.

### SAP HANA Security Guide

#### 7.5.5 Changing Users

You can change a user’s authentication information, grant them new privileges and roles, as well as revoke previously granted privileges and roles.

To be able to change a user’s properties (that is, password, authentication information), you must have the system privilege USER ADMIN.

**Note:**

A user can change his or her own password without USER ADMIN.

If you are implementing external authentication, the following prerequisites apply:
• If you are implementing Kerberos user authentication, you have installed the relevant Kerberos client software from your operating system software package and configured the SAP HANA database accordingly.

• If you are implementing Security Assertion Markup Language (SAML) user authentication, you have created the necessary identity providers.

To be able to change the user’s privileges, you must have the privileges required to grant and revoke privileges and roles to and from the user.

1. In the Navigator view, choose Security Users.
2. Open the relevant user and make the required changes.
   You can change the following:
   ○ Authentication methods supported for the user
   ○ Password for internal authentication
   ○ External ID for Kerberos authentication, that is the user principal name (UPN) specified in the Microsoft Active Directory or the Kerberos Key Distribution Center
   ○ Identity provider and external user ID for SAML authentication
   ○ Granted roles and privileges (object privileges, analytic privileges, system privileges)
   ○ Whether or not the user is allowed to pass on his or her privileges to other users (Grantable to other users and roles option)

   Note: This option is not available if you are granting either a role that was created in the repository or a privilege on a repository object.

3. Choose the (Deploy) button to save the changes.
   If you changed the user’s password (internal authentication), the system verifies that it complies with the system’s password policy.

Related Links
Prerequisites for Granting and Revoking Privileges and Roles [page 57]
To be able to grant and revoke privileges and roles to and from users and roles, several prerequisites must be met.

Configuring Kerberos for SAP HANA Database Hosts [page 44]
SAP HANA supports Kerberos version 5 for single sign-on based on Active Directory (Microsoft Windows Server) or Kerberos authentication servers. Both ODBC database clients and JDBC database clients support the Kerberos protocol.

Configuring SAML Identity Providers [page 45]
If you are implementing Security Assertion Markup Language (SAML) user authentication, you must configure SAML identity providers for the required users.

Working with the Password Blacklist [page 43]
When creating new passwords, users are not allowed to use blacklisted words or partial words. Use SQL commands to insert or delete words or partial words from the password blacklist table.

SAP HANA Security Guide

7.5.6 Deleting Users
You may need to delete a database user if an employee leaves your organization for example.

To delete a user, you must have the system privilege USER ADMIN.
1. In the Navigator view, choose Security Users.

2. Right-click the relevant user and choose Delete.

3. Confirm whether or not it acceptable that dependent objects, such as tables, views, and procedures, are deleted with the user (Cascade option).

**Caution:** If you choose the Cascade option, all dependent objects are deleted and no longer available.

### 7.5.7 Deactivating Users

Users can be automatically deactivated for security reasons, for example, if they violate password policy rules. However, as a database administrator, you may need to explicitly deactivate a user, for example, if an employee temporarily leaves the company or if a security violation is detected.

To be able to deactivate a user, you must have the system privilege USER ADMIN.

1. In the Navigator view, choose Security Users and open the relevant user.

2. From the editor toolbar, choose (Deactivate User...)

The database user is now deactivated and remains so until you reactivate. The user still exists in the database, but cannot connect to the database any more. The reason (explicit deactivation) and the time of deactivation are displayed in the user’s details.

### 7.5.8 Reactivating Users

As a database administrator, you may need to reactivate a user, for example, you explicitly deactivated the user or the user has made too many invalid log-on attempts.

To be able to reactivate a user, you must have the system privilege USER ADMIN.

1. In the Navigator view, choose Security Users and open the relevant user.

2. From the editor toolbar, choose (Activate User...)
   
   You are prompted to enter a new password for the user. The user is now reactivated.

### 7.5.9 Verifying Users' Privileges

You can see which privileges and roles a user has been granted directly in their user definition. However, if privileges are granted indirectly through hierarchical roles, you can query several system views to get detailed information about exactly which privileges and roles a user has and how they come to have them.

To verify users' privileges, you must have the system privilege CATALOG READ or CATALOG ADMIN.

1. Open the SQL console.

2. Depending on the information required, execute one or more of the following statements:

<table>
<thead>
<tr>
<th>Information Required</th>
<th>SQL Statement</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directly granted privileges</td>
<td><code>SELECT * FROM &quot;PUBLIC&quot;.&quot;GRANTED_PRIVILEGES&quot; where grantee = '&lt;USER&gt;'</code></td>
<td>Privileges granted directly to the specified user (or role) are listed. Privileges contained within granted roles are not shown.</td>
</tr>
</tbody>
</table>
### Information Required

<table>
<thead>
<tr>
<th>SQL Statement</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>where grantee = '&lt;USER&gt;' with where grantee = '&lt;ROLE&gt;'</td>
<td>All roles granted directly to the specified user (or role) are listed. Roles contained within granted roles are not shown.</td>
</tr>
<tr>
<td>SELECT * FROM &quot;PUBLIC&quot;.&quot;GRANTED_ROLES&quot; where grantee = '&lt;USER/ROLE_NAME&gt;'</td>
<td>All privileges granted to the specified user both directly and indirectly through roles are listed separately.</td>
</tr>
<tr>
<td>SELECT * FROM &quot;PUBLIC&quot;.&quot;EFFECTIVE_PRIVILEGES&quot; where user_name = '&lt;USER&gt;'</td>
<td>All roles granted to the specified user both directly and indirectly through other roles are listed separately.</td>
</tr>
</tbody>
</table>

### Directly granted roles

**Note:** It is possible to query the roles directly granted to a role by replacing where grantee = '<USER>' with where grantee = '<ROLE>'

### Directly and indirectly granted privileges

### Directly and indirectly granted roles

### Related Links

*Opening the SQL Console* [page 18]

Some tasks may require you to work with SQL statements, for example, certain administration tasks can only be performed using SQL. You can enter, execute, and analyze SQL statements in the SQL console.

*SAP HANA System Tables and Monitoring Views Reference*

## 7.5.10 Prerequisites for Granting and Revoking Privileges and Roles

To be able to grant and revoke privileges and roles to and from users and roles, several prerequisites must be met.

<table>
<thead>
<tr>
<th>Action</th>
<th>Prerequisite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grant system privilege, object privilege, or package privilege to user or role</td>
<td>Granting user must have the privilege being granted and be authorized to grant it to other users and roles</td>
</tr>
<tr>
<td>Grant analytic privilege to user or role</td>
<td>Granting user must have the object privilege EXECUTE on the procedure <code>GRANT_ACTIVATED_ANALYTICAL_PRIVILEGE</code></td>
</tr>
<tr>
<td>Grant object privilege on activated modeled objects, such as calculation views, to user or role</td>
<td>Granting user must have the object privilege EXECUTE on the procedure <code>GRANT_PRIVILEGE_ON_ACTIVATED_CONTENT</code></td>
</tr>
<tr>
<td>Grant role created in runtime to user or role</td>
<td>Granting user must have the role being granted and be authorized to grant it to other users and roles, or</td>
</tr>
<tr>
<td>Action</td>
<td>Prerequisite</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Grant role created in the repository to user or role</td>
<td>Granting user must have the system privilege ROLE ADMIN</td>
</tr>
<tr>
<td>Grant role created in the repository to user or role</td>
<td>Granting user must have the object privilege EXECUTE on the procedure GRANT_ACTIVATED_ROLE</td>
</tr>
<tr>
<td>Grant application privilege</td>
<td>Granting user must have the object privilege EXECUTE on the procedure GRANT_APPLICATION_PRIVILEGE</td>
</tr>
<tr>
<td>Grant object privilege on schema containing activated modeled objects, such as calculation views, to user or role</td>
<td>Granting user must have the object privilege EXECUTE on the procedure GRANT_SCHEMA_PRIVILEGE_ON_ACTIVATED_CONTENT</td>
</tr>
<tr>
<td>Revoke system privilege, object privilege, or package privilege from user or role</td>
<td>Revoking user must be the user who granted the privilege</td>
</tr>
<tr>
<td>Revoke object privilege on schema activated containing modeled objects, such as calculation views, from user or role</td>
<td>Revoking user must have the object privilege EXECUTE on the procedure REVOKE_SCHEMA_PRIVILEGE_ON_ACTIVATED_CONTENT</td>
</tr>
<tr>
<td>Revoke analytic privilege from user or role</td>
<td>Revoking user must have the object privilege EXECUTE on the procedure REVOKE_ACTIVATED_ANALYTICAL_PRIVILEGE</td>
</tr>
<tr>
<td>Revoke role created in runtime from user or role</td>
<td>Revoking user must be the user who granted the role</td>
</tr>
<tr>
<td>Revoke role created in the repository from user or role</td>
<td>Revoking user must have the object privilege EXECUTE on the procedure REVOKE_ACTIVATED_ROLE</td>
</tr>
<tr>
<td>Revoke application privilege</td>
<td>Revoking user must have the object privilege EXECUTE on the procedure REVOKE_APPLICATION_PRIVILEGE</td>
</tr>
<tr>
<td>Revoke object privilege on activated modeled objects, such as calculation views from user or role</td>
<td>Revoking user must have the object privilege EXECUTE on the procedure REVOKE_PRIVILEGE_ON_ACTIVATED_CONTENT</td>
</tr>
</tbody>
</table>

### 7.6 Disabling Default User Filtering of Schemas

In the Navigator view, the schemas available in the catalog of a system are filtered according to user by default. If, as a database administrator, you need to see all available schemas, you must disable the default schema filter.

The schemas available in the catalog of a system are filtered according to user by default. This is because in large SAP HANA systems hosting multiple applications with hundreds, thousands, maybe even hundreds of thousands of users, it would be impossible for individual users to identify the schemas with which they are permitted to work if all schemas were visible. Therefore, the connected user sees only those schemas for which at least one of the following criterion applies:
- The user is the schema owner.
- The user has at least one privilege on the schema.
- The user has at least one privilege on at least one object in the schema.
- The user owns at least one object in the schema.

**Note:** For all privilege checks, not only privileges directly granted to the user but also privileges granted to one of his or her roles (or to roles in these roles) are considered.

As a result, users with DATA ADMIN or CATALOG READ privilege, in particular user SYSTEM, do not see all available schemas. If, as a database administrator, you need to see all available schemas, you must disable the default schema filter.

1. In the **Navigator** view, right-click **Catalog** and choose **Filters...**
   The **Filter for Schema** dialog box opens.
2. Select **Display all schemas**.
3. Optional: Specify a filter pattern to reduce the number of schemas displayed.
   This is useful if the total number of schemas exceeds the number of displayable items in the tree (configured under **Preferences > Catalog**). If this is the case, then you will not see all schemas at once and will have to browse.
4. Choose **OK**.

All schemas are displayed filters according the specified filter pattern.
8  Monitoring SAP HANA Systems

8.1  The System Monitor

The System Monitor provides you with an overview of all your SAP HANA systems at a glance. You can drill down into the details of an individual system in the Administration editor.

To open the System Monitor, in the toolbar of the Navigator view choose the (System Monitor) button. The following information is displayed:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System ID</td>
<td>ID assigned to system when added</td>
</tr>
<tr>
<td>Operational State</td>
<td>Indication of the system status</td>
</tr>
<tr>
<td></td>
<td>The following statuses are possible:</td>
</tr>
<tr>
<td></td>
<td>● All services have started ( )</td>
</tr>
<tr>
<td></td>
<td>● All services have started but the license for the system is invalid or has expired ( )</td>
</tr>
<tr>
<td></td>
<td>● Some services have not started ( )</td>
</tr>
<tr>
<td></td>
<td>● Some services are still in the process of starting ( )</td>
</tr>
<tr>
<td>Alerts</td>
<td>The statistics server generates alerts for the system when resource usage and statistical thresholds are violated. These alerts are categorized as low, medium, or high priority. The number of alerts and their status is shown here.</td>
</tr>
<tr>
<td>Data Disk (GB)</td>
<td>The size of the data volume on disk</td>
</tr>
<tr>
<td>Log Disk (GB)</td>
<td>The size of the log volume on disk</td>
</tr>
<tr>
<td>Trace Disk (GB)</td>
<td>The size of trace files on disk</td>
</tr>
<tr>
<td>Database Resident Memory (GB)</td>
<td>The size of resident memory at operating system level owing to SAP HANA database processes</td>
</tr>
<tr>
<td>System Resident Memory (GB)</td>
<td>The total size of resident memory in the operating system</td>
</tr>
<tr>
<td>Used Memory (GB)</td>
<td>The amount of physical memory used by the SAP HANA database</td>
</tr>
<tr>
<td>CPU (%)</td>
<td>Percentage of CPU used by the SAP HANA database</td>
</tr>
<tr>
<td>Hostname</td>
<td>The name of the server hosting the SAP HANA database</td>
</tr>
<tr>
<td>Instance Number</td>
<td>The instance number is the administrative unit that comprises the server software components</td>
</tr>
<tr>
<td>System Data Disk (GB)</td>
<td>Total disk space occupied on disk(s) containing data</td>
</tr>
</tbody>
</table>
### Column Description

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Log Disk (GB)</td>
<td>Total disk space occupied on disk(s) containing log files</td>
</tr>
<tr>
<td>System Trace Disk (GB)</td>
<td>Total disk space occupied on disk(s) containing trace files</td>
</tr>
<tr>
<td>System Physical Memory (GB)</td>
<td>Total amount of physical memory used</td>
</tr>
<tr>
<td>System CPU (%)</td>
<td>Overall CPU usage</td>
</tr>
<tr>
<td>Distributed</td>
<td>Indicates whether the system is running on a single host or it is a distributed system running on more than one host</td>
</tr>
<tr>
<td>Start Time First</td>
<td>Time that the first service started</td>
</tr>
<tr>
<td></td>
<td>This value is updated when system is restarted for any reason.</td>
</tr>
<tr>
<td>Start Time Latest</td>
<td>Time that the last service was started, if, for example, one of the services was re-started individually</td>
</tr>
<tr>
<td>Version</td>
<td>The software version number of the SAP HANA studio</td>
</tr>
<tr>
<td>Platform</td>
<td>Operating system on which the SAP HANA studio is running</td>
</tr>
</tbody>
</table>

### Configuration Options

The toolbar of the System Monitor in the top-right corner provides you with the following functions:

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Filter" /></td>
<td>Allows you to select a sub-set of systems to display in the System Monitor, if for example you have a very large number of systems</td>
</tr>
<tr>
<td><img src="image" alt="Properties" /></td>
<td>Allows you to configure properties of the System Monitor, such as the refresh interval and whether or not you want it to open automatically when you open the SAP HANA studio</td>
</tr>
<tr>
<td><img src="image" alt="Configure Viewer" /></td>
<td>Allows you to configure which information is displayed, that is, which columns are visible</td>
</tr>
</tbody>
</table>

### Related Links

*Monitoring in the Administration Editor* [page 61]

To identify problems with the SAP HANA database early and avoid disruptions, you need to monitor your systems continuously.

### 8.2 Monitoring in the Administration Editor

To identify problems with the SAP HANA database early and avoid disruptions, you need to monitor your systems continuously.

While the System Monitor provides you with an overview of all your systems at a glance, the Administration editor allows you to drill down into the details of resource usage and performance for each system.
In the Administration editor, you can monitor the following:

- Overall system state and resource usage by system and host
- Status and resource usage of all system components, for example, name server, index server, statistics server, and so on
- Failover status and configuration of hosts in distributed systems
- Alerts issued by the statistics server in relation to system status, performance, and resource consumption
- Disk space consumed by system processes for the various storage types (data, log, and trace)
- System performance, for example, by analyzing performance indicators such as expensive statements, running threads, and load history

Related Links

The System Monitor [page 60]
The System Monitor provides you with an overview of all your SAP HANA systems at a glance. You can drill down into the details of an individual system in the Administration editor.

Opening the Administration Editor [page 17]
The Administration editor is available in the Administration Console perspective and is the main tool for performing administration and monitoring activities.

Opening the SQL Console [page 18]
Some tasks may require you to work with SQL statements, for example, certain administration tasks can only be performed using SQL. You can enter, execute, and analyze SQL statements in the SQL console.

8.3 Monitoring Overall System Status

When you open the Administration editor for a particular SAP HANA system, the Overview tab provides you with a summary of the overall status of the system, as well as an overview of resource usage.

Resource usage values are presented in such a way that you can compare the SAP HANA system with the operating system as a whole. If the system is distributed, resource usage values are aggregated across all hosts. An additional bar shows the host with the highest (most critical) resource usage.

The bars indicating resource usage (memory, CPU, and disk) change color (green, yellow, and red) based on configurable check thresholds.

The following figure shows an example of the system overview of a distributed system.

Figure 2: Distributed System Overview in the Administration Editor
The following information is available:

### Screen Area

<table>
<thead>
<tr>
<th>Information Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Information</td>
</tr>
<tr>
<td>● General information about the SAP HANA system, such as operational status, whether or not the system is distributed, the number of hosts (if distributed), and database version</td>
</tr>
<tr>
<td>● The status of replication from your productive system to a secondary system</td>
</tr>
<tr>
<td>This information is only available and applicable if you are operating a secondary instance of your database (for example, in a high availability scenario). If this is the case, then content from the primary or productive instance of your database is replicated to the secondary instance. More detailed information on this replication status is available on the Secondary System Status tab.</td>
</tr>
</tbody>
</table>

### Alerts and Messages

<table>
<thead>
<tr>
<th>Priority-rated alerts and messages reported by the statistics server</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Used Memory</td>
</tr>
<tr>
<td>The following key indicators of memory usage are displayed:</td>
</tr>
<tr>
<td>● <strong>Used Memory</strong></td>
</tr>
<tr>
<td>The total amount of memory currently in use by SAP HANA is referred to as its used memory.</td>
</tr>
<tr>
<td>● <strong>Peak Used Memory</strong></td>
</tr>
<tr>
<td>The Used Memory value is a current measurement. The Peak Used Memory value is the highest used memory value recorded. This is useful for keeping track of the maximum value for used memory over time.</td>
</tr>
</tbody>
</table>

**Note:** Peak used memory is a resettable value. This can be useful if you want to establish the impact of a certain workload on memory usage. So for example, you can reset peak used memory, run the workload, and then examine the new peak used memory value. You can reset...
peak used memory on the Landscape > Services tab. From the context menu, choose Reset Memory Statistics.

- **Allocation Limit**

  The SAP HANA system, across its different processes, reserves a pool of memory before actual use. This pool of allocated memory is preallocated from the operating system over time, up to a predefined allocation limit, and is then efficiently used as needed by the SAP HANA database code. More memory is allocated to the pool as memory consumption increases. If the amount of memory used nears the allocation limit, SAP HANA may run out of memory if it cannot free memory.

<table>
<thead>
<tr>
<th>Screen Area</th>
<th>Information Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resident Memory</td>
<td>Resident memory is the amount of physical memory that is actually being used from the perspective of the operating system.</td>
</tr>
<tr>
<td></td>
<td>It is possible that the Used Memory value is lower than the Database Resident value if SAP HANA returns memory back to its memory pool (for example, after a temporary computation) and does not inform the operating system. This is normal.</td>
</tr>
<tr>
<td>CPU Usage</td>
<td>The information displayed here indicates the percentage of CPU used by the SAP HANA system compared with the operating system as a whole.</td>
</tr>
<tr>
<td>Disk Usage</td>
<td>The information displayed here indicates disk space occupied by data, log, and trace files belonging to the SAP HANA system compared with the operating system as a whole.</td>
</tr>
</tbody>
</table>

**Related Links**

- **Monitoring Memory Usage** [page 92]
  Memory is a fundamental resource of the SAP HANA database. Understanding how the SAP HANA database requests, uses, and manages this resource is crucial to the understanding of SAP HANA.

- **Monitoring Disk Space** [page 88]
  To ensure that the database can always be restored to its most recent committed state, you must ensure that there is enough space on disk for data and log volumes. You can monitor disk usage, volume size, and other disk activity statistics on the Volumes tab of the Administration editor.

- **Monitoring Alerts** [page 73]
  As the administrator of an SAP HANA system, you actively monitor the status of the system, its services, and the consumption of system resources. However, you are also alerted of critical situations, for example: a disk is becoming full, CPU usage is reaching a critical level, or a server has stopped.

- **Configuring Check Thresholds** [page 83]
  For some checks performed by the statistics server, you can configure when an alert is issued, that is the alert condition. A check can have a low, medium, and high priority threshold.
8.4 Monitoring System Status and Resource Usage

An SAP HANA system comprises all the system components of an installation of SAP HANA, for example, the index server, name server, and statistics server. You can monitor the operational status and resource usage of individual system components on the [Landscape > Services] tab of the Administration editor.

Overview of SAP HANA System Architecture

An SAP HANA system comprises all the system components of an installation of SAP HANA. In a single-host system, all these components are installed on a single SAP HANA database instance on one host. In a distributed SAP HANA system, they are installed on multiple database instances on different hosts. The following is a brief overview of the system components of the SAP HANA system and the corresponding OS processes.

![Note](Note: The SAP HANA system consists of a number of services, which, from the perspective of the OS, appear as processes. In a distributed system, not all services are necessarily running on all instances. For example, only one instance in the system hosts the statistics server so the statisticsserver service runs only on one host.)

<table>
<thead>
<tr>
<th>System Component</th>
<th>OS Process</th>
<th>Service Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index server</td>
<td>hdbindexserver</td>
<td>indexserver</td>
<td>The index server contains the actual data stores and the engines for processing the data.</td>
</tr>
<tr>
<td>Preprocessor server</td>
<td>hdbpreprocessor</td>
<td>preprocessor</td>
<td>The preprocessor server is used by the index server to analyze text data and extract the information on which the text search capabilities are based.</td>
</tr>
<tr>
<td>Name server</td>
<td>hdbnameserver</td>
<td>nameserver</td>
<td>The name server owns the information about the topology of the SAP HANA system. In a distributed system with instances of the SAP HANA database on multiple hosts, the name server knows where the components are running and which data is located on which server.</td>
</tr>
<tr>
<td>Statistics server</td>
<td>hdbstatisticsserver</td>
<td>statisticsserver</td>
<td>This statistics server collects information about status, performance and resource consumption from all components belonging to the SAP HANA system. Monitoring clients such as the SAP HANA studio access the statistics server to get the status of various alert monitors. The</td>
</tr>
</tbody>
</table>
The statistics server also provides a history of measurement data for further analysis.

### SAP HANA XS
- **OS Process**: hdbxsengine
- **Service Name**: xsengine
- **Description**: SAP HANA Extended Application Services (SAP HANA XS) provides applications and application developers with access to the SAP HANA database using a consumption model that is exposed via HTTP. In addition to providing application-specific consumption models, SAP HANA XS also host system services that are part of the SAP HANA database, for example: search services and a built-in Web server that provides access to static content stored in the SAP HANA repository.

### SAP Start service
- **OS Process**: sapstartsrv
- **Service Name**: sapstartsrv
- **Description**: sapstartsrv is responsible for starting and stopping the other services in the correct order. It also performs other functions, such as monitoring their runtime state.

## Process Status and Resource Usage

You can monitor the operational status of the individual components of your system on the Services tab of the Administration editor.

Here, you can see the status of the services that start when the system is started. The initial connection to the system is established by the sapstartsrv service. If you have a distributed system, the services that start depend on which components are actually installed on the instance. For each operating system process, you can also see detailed information about its memory consumption. This allows you to get a more detailed breakdown of resource usage and troubleshoot performance bottlenecks.

The available filters allow you to show and hide the information according to host and/or service. This is generally only useful if you have a distributed system.

The following information is displayed on Services sub-tab by default. You can configure the view by choosing the (Configure Viewer) button. For example, several additional columns are available.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Indicates the status of the service</td>
</tr>
<tr>
<td></td>
<td>The following statuses are possible:</td>
</tr>
</tbody>
</table>

© 2012 SAP AG. All rights reserved.
<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>● The service is started (聞き)</td>
<td>The daemon service displays △ while the host or any if its services are starting or stopping.</td>
</tr>
<tr>
<td>● The service is stopping or starting (△)</td>
<td></td>
</tr>
<tr>
<td>● The service is stopped (●)</td>
<td></td>
</tr>
<tr>
<td>Host</td>
<td>Name of the host on which the service is running</td>
</tr>
<tr>
<td>Port</td>
<td>Port that the system uses for internal communication between services</td>
</tr>
<tr>
<td>Service</td>
<td>Service name, for example, indexserver, nameserver, statisticsserver, xsengine, and so on</td>
</tr>
<tr>
<td>Detail</td>
<td>Role of the host on which the service is running</td>
</tr>
<tr>
<td>● Master</td>
<td>The host is the active master host.</td>
</tr>
<tr>
<td>● &lt;Empty&gt;</td>
<td>The host is an active slave host.</td>
</tr>
<tr>
<td>● Standby</td>
<td>The host is a standby host.</td>
</tr>
<tr>
<td>Start Time</td>
<td>Time at which the service started</td>
</tr>
<tr>
<td>Process ID</td>
<td>Process ID of the OS process</td>
</tr>
<tr>
<td>CPU</td>
<td>Bar view showing the CPU usage of the service</td>
</tr>
<tr>
<td>Memory</td>
<td>Bar view showing the used memory of the service in relation to physical memory available and the effective allocation limit of the service</td>
</tr>
<tr>
<td>Used Memory (MB)</td>
<td>Amount of memory currently used by the service</td>
</tr>
<tr>
<td>Peak Used Memory (MB)</td>
<td>Highest amount of memory ever used by the service</td>
</tr>
<tr>
<td>Effective Allocation Limit (MB)</td>
<td>Effective maximum memory pool size that is available to the process considering the current memory pool sizes of other processes</td>
</tr>
</tbody>
</table>
Memory is a fundamental resource of the SAP HANA database. Understanding how the SAP HANA database requests, uses, and manages this resource is crucial to the understanding of SAP HANA.

### 8.5 Monitoring Host Status and Failover Configuration

The SAP HANA database supports high availability in a distributed system by providing for host failover. If an active host fails, for example, because of a hardware failure, standby hosts can take over and thus ensure the continued availability of the database.

You can monitor the status of individual hosts on the **Landscape Configuration** tab. Here, you can see all the hosts in the system, whether or not they are operational, as well as additional information about their failover status and configuration.

The following table provides an explanation of the information displayed.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>Host name</td>
</tr>
<tr>
<td>Active</td>
<td>Indicates the status of services running on the host</td>
</tr>
<tr>
<td></td>
<td>The following statuses are possible:</td>
</tr>
<tr>
<td></td>
<td>● YES (✓)</td>
</tr>
<tr>
<td></td>
<td>All services are active.</td>
</tr>
<tr>
<td></td>
<td>● PARTIAL (✓)</td>
</tr>
<tr>
<td></td>
<td>Some services active.</td>
</tr>
<tr>
<td></td>
<td>● STARTING (△)</td>
</tr>
<tr>
<td></td>
<td>Some services are active, some are starting.</td>
</tr>
<tr>
<td></td>
<td>● STOPPING (△)</td>
</tr>
<tr>
<td></td>
<td>Some services are active, some are stopping.</td>
</tr>
</tbody>
</table>

**Note:** You can configure the view by choosing the **Configure Viewer** button. You can also call up information about the meaning of the various statuses by choosing the **Display Information** button.
### Column Description

- **NO (●)**
  No services active

### Host Status

Indicates whether or not the system is operational and the host’s status.

The following statuses are possible:

- **OK (■)**
  The system is operational and the host’s actual role corresponds to its configured role.

- **IGNORE (◇)**
  The system is operational. The host is configured as a standby host and is available, but not in use.

- **INFO (ℹ)**
  The system is operational. The host’s actual role is different from its configured role.

- **WARNING (▲)**
  The system is not operational. The host will become available after start-up or failover.

- **ERROR (☒)**
  The system is not operational. The host is missing.

### Failover Status

Displays the failover status so you can see which hosts are active and which are on standby.

The following statuses are possible:

- **<Empty>**
  Failover is neither active nor pending.

- **WAITING ... SEC**
  The host has failed. The system is waiting to fail over.

- **WAITING**
  The host has failed. The system is waiting for the host to restart to prevent unnecessary failover.

- **FAILOVER TO <host>**
  The host has failed and failover to a target host is in progress.

- **FAILBACK TO <host>**
  Failback to a worker host is in progress. This happens when the assigned standby host is stopped. However, there is no automatic failback while the standby host is still assigned since this would cause downtime.

- **FAILED**
## Nameserver Role (Configured)

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies the host’s configured role as name server</td>
</tr>
<tr>
<td>The following roles are possible:</td>
</tr>
<tr>
<td>- MASTER 1, MASTER 2, MASTER 3</td>
</tr>
<tr>
<td>When you install a distributed system, up to three hosts are automatically configured as master name servers. The configured nameserver role of these hosts is MASTER 1, MASTER 2, and MASTER 3.</td>
</tr>
<tr>
<td>- SLAVE</td>
</tr>
<tr>
<td>Additional hosts in your system are configured as slave name servers. The configured nameserver role of these hosts is SLAVE.</td>
</tr>
<tr>
<td><strong>Note:</strong> To ensure that the SAP HANA studio can connect to the database, it is recommended that you specify these 3 hosts in the connection properties of the system because at least one of them will be active. You can do this by right-clicking the system in the Navigator view and choosing Properties Database User Logon Additional Properties.</td>
</tr>
</tbody>
</table>

## Nameserver Role (Actual)

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies the host’s actual role as name server</td>
</tr>
<tr>
<td>The following roles are possible:</td>
</tr>
<tr>
<td>- MASTER</td>
</tr>
<tr>
<td>During system start-up, one of the hosts configured as master name servers (that is, those hosts with configured name server role MASTER 1, MASTER 2, or MASTER 3) is designated to be the active master name server. The actual nameserver role of this host is MASTER. This master name server assigns one volume to each starting index server (those with actual role MASTER or SLAVE), or no volume if it is a standby host (actual indexserver role STANDBY).</td>
</tr>
<tr>
<td>If this active master nameserver host fails, one of the remaining hosts configured as a master name server becomes the active master name server.</td>
</tr>
<tr>
<td>- SLAVE</td>
</tr>
<tr>
<td>The actual nameserver role of the remaining hosts configured as master and slave hosts is SLAVE.</td>
</tr>
</tbody>
</table>

## Indexserver Role (Configured)

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies the host’s configured role as index server</td>
</tr>
<tr>
<td>The following roles are possible:</td>
</tr>
<tr>
<td>- WORKER</td>
</tr>
<tr>
<td>- STANDBY</td>
</tr>
<tr>
<td>When you install a distributed system, you can configure hosts either as WORKER or STANDBY index servers. A host configured as a standby index server is not used for database processing. All database processes run on the standby host, but they are idle and do not allow SQL connections.</td>
</tr>
</tbody>
</table>
### Column: Indexserver Role (Actual)

Specifies the host's actual role as index server.

The following roles are possible:

- **MASTER**
  
  The actual master index server is assigned on the same host as the name server with the actual role MASTER. The actual index server role of this host is MASTER. The master index server provides metadata for the other active index servers (that is, those with actual indexserver role SLAVE).

- **SLAVE**
  
  The actual index server role of remaining hosts (except those configured as standby hosts) is SLAVE. These are active index servers and are assigned to one volume. If an active index server fails, the active master name server assigns its volume to one of the standby hosts.

- **STANDBY**
  
  The actual indexserver role of standby hosts is STANDBY. A standby host is not assigned a volume by the active master name server and it does not open an SQL port.

During normal operation when all hosts are available, a host with the configured role WORKER has the actual role MASTER or SLAVE, and a host with the configured role STANDBY has the actual role STANDBY. In the event of failover, the actual index server role of a host with the configured role STANDBY changes to SLAVE. The host status of the failed host changes from OK to INFO and the host status of the standby host changes from IGNORE to INFO.

**Note:** Failover is configured only for the name server and the index server on each host. The other components (for example, statistics server) are not configured individually as they are always failed over together with the index server.

### Column: Failover Group (Configured/Actual)

A failover group can be defined for each host. In the event of failover, the name server tries to fail over to a host within the same group.

### Column: Storage Partition

Specifies the number of the mnt000... sub-directory used by the host for storing data and logs, for example, 1 if the sub-directory is mnt00001, 2 if it is mnt00002, and so on.

During installation, volumes for storing data and log files are defined. These are the directories where data and logs are stored. The default directories are:

- `/usr/sap/<SID>/global/hdb/data` for data
- `/usr/sap/<SID>/global/hdb/log` for logs

Each active host has exactly one sub-directory beneath these directories called mnt00001, mnt00002, and so on. The next level in the file hierarchy is the actual volume, with one sub-directory for each service called hdb00001, hdb00002, and so on.

In the event of failover, the volumes of the failed host are re-assigned to the standby host.

### Column: Removal Status

Indicates the status of the table redistribution operation used to move data off the index server of a host that you plan to remove.
Before you can remove an active host from your system, you must move the tables on the index server of this host to the index servers on the remaining hosts in the system. You can do this by right-clicking the host and choosing Remove Host... Once the value in the Removal Status column changes to REORG FINISHED or REORG NOT REQUIRED, you can physically remove the host using the SAP HANA On-Site Configuration tool.

The following statuses are possible:

- <Empty>
  Host has not been marked for removal.
- REORG PENDING
  A redistribution operation is required to move tables to other hosts.
- REORG ACTIVE
  A redistribution operation is in progress. For more information, you can query the system tables SYS.REORG_OVERVIEW and SYS.REORG_STEPS.
- REORG FAILED
  A redistribution operation was executed and failed. For more information, query the system table SYS.REORG_STEPS.
- REORG FINISHED
  A redistribution operation has completed. The host can be uninstalled.
- REORG NOT REQUIRED
  A redistribution operation not required. The host can be uninstalled.

Example:

Table 9: Typical Configuration for a Distributed System

<table>
<thead>
<tr>
<th>Host</th>
<th>Name Server (Configured Role)</th>
<th>Name Server (Actual Role)</th>
<th>Index Server (Configured Role)</th>
<th>Index Server (Actual Role)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial host</td>
<td>Master 1</td>
<td>Master</td>
<td>Worker</td>
<td>Master</td>
</tr>
<tr>
<td>1st host added</td>
<td>Master 2</td>
<td>Slave</td>
<td>Worker</td>
<td>Slave</td>
</tr>
<tr>
<td>2nd host added</td>
<td>Master 3</td>
<td>Slave</td>
<td>Worker</td>
<td>Slave</td>
</tr>
<tr>
<td>3rd host added</td>
<td>Slave</td>
<td>Slave</td>
<td>Worker</td>
<td>Slave</td>
</tr>
<tr>
<td>4th host added</td>
<td>Slave</td>
<td>Slave</td>
<td>Worker</td>
<td>Slave</td>
</tr>
<tr>
<td>5th host added</td>
<td>Slave</td>
<td>Slave</td>
<td>Worker</td>
<td>Slave</td>
</tr>
<tr>
<td>6th host added</td>
<td>Slave</td>
<td>Slave</td>
<td>Worker</td>
<td>Slave</td>
</tr>
<tr>
<td>7th host added</td>
<td>Slave</td>
<td>Slave</td>
<td>Standby</td>
<td>Standby</td>
</tr>
</tbody>
</table>

Note: Host roles for failover are normally configured during installation. The options available to you on the Configuration tab when you choose the Configure Hosts for Failover Situation button are limited.
You can only switch the configured roles of hosts; you cannot increase or decrease the number of worker hosts and standby hosts in relation to each other.

The primary reason for changing the configured roles in the Configure Hosts for Failover Situation dialog box is to prepare for the removal of a host. In this case, you would change the configured role of the name server host to SLAVE and the configured role of the index server host to STANDBY before stopping the database instance on the host and removing the host.

Related Links

About High Availability [page 196]

Redistributing Tables Before Removing a Host [page 145]

Before you can remove a host from your SAP HANA system, you must move the tables on the index server of the host in question to the index servers on the remaining hosts in the system.

SAP HANA Installation Guide with Unified Installer

8.6 Monitoring Alerts

As the administrator of an SAP HANA system, you actively monitor the status of the system, its services, and the consumption of system resources. However, you are also alerted of critical situations, for example: a disk is becoming full, CPU usage is reaching a critical level, or a server has stopped.

The statistics server is the main component of the monitoring infrastructure of the SAP HANA database. It collects information about status, performance, and resource usage from all components of the SAP HANA database. When an alert condition for a particular measurement is fulfilled, the statistics server issues an alert. The priority of the alert indicates the severity of the problem and depends on the nature of the check and configured threshold values. For example, if 90% of available disk space is used, a low priority alert is issued; if 98% is used, a high priority alert is issued. The SAP HANA studio can access measurement and alert data from the statistics server and in this way warn you of potential problems.

Note: The statistics server also provides a history of performance and resource data for further analysis. You can access this data by viewing or querying the content of the tables in the _SYS_STATISTICS schema.

By default, when an SAP HANA system is started, the statistics server is automatically started on the host of the active master name server. The statistics server internally uses SQL statements to collect information from all index servers. Important alerts from the information collected on the system state are summarized on the Overview tab of the Administration editor and displayed in detail on the Alerts tab.

Note: To ensure that you are seeing the latest information, refresh the Administration editor regularly.

Alert Display

When you open the Alerts tab, all current, unresolved alerts are displayed by default. To see all alerts, choose the corresponding entry from the Show drop-down list. Alerts are displayed according to the following time periods:
<table>
<thead>
<tr>
<th>Time Period</th>
<th>Alerts Displayed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last 15 minutes, last 30 minutes, last hour, last 2 hours, and today</td>
<td>Alerts generated in the corresponding time period are shown. If an alert was generated 10 minutes ago, it appears under all these headings.</td>
</tr>
<tr>
<td>Yesterday</td>
<td>Only alerts that were generated yesterday are shown.</td>
</tr>
<tr>
<td>Last week</td>
<td>Only alerts generated during the previous week (Sunday to Saturday) are shown.</td>
</tr>
<tr>
<td>Two weeks ago, and so on</td>
<td>Only alerts generated during that week are displayed.</td>
</tr>
</tbody>
</table>

1. **Note:** Alerts are not rolled over into the following weeks. This enables you to compare the performance of the system over selected periods, as well as to view the alerts.

You can refine the list of displayed alerts further by specifying a filter parameter as follows:

- To filter according to a specific priority or timeframe, choose the `(Set Alert Filter)` button in the toolbar on the top-right of the tab and select the required filter(s).
- To filter according to a specific word in the check description, enter the word in the `Filter` field (for example, `license`).

**Detailed Alert Information**

You can view the detailed information about an alert by double-clicking it. The `Alert Details` dialog box appears with information including:

- A full description of the alert
- The time stamp for this occurrence of the alert
- Information about how to resolve the alert
- A history of when this alert was generated in the past

The `Copy` button in the `Alert Details` dialog box allows you to copy the details of the alert to the clipboard, including its time(s) of occurrence. Note that only the 10 most recent occurrences are copied. Further occurrences are indicated by an ellipsis (…)

**Check Information**

The `Check Information` area at the bottom of the `Alerts` tab provides a full list of all the checks carried out by the statistics server. In addition to a description of what each check does (`Description`) and what you need to do in the event of an alert (`User Action`), you can see important scheduling information:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Priority</td>
<td>The most severe alert that was generated the last time the check was carried out</td>
</tr>
</tbody>
</table>
For checks that consider only one object (for example, the check that determines how many days until your license expires), only one alert can be generated. This is automatically the most severe.

However, for checks that consider several objects (for example, the check that determines CPU utilization in a system with multiple hosts), several alerts can be generated for one check. The most severe is recorded here. For example, if in a distributed system with 5 hosts, the CPU utilization of 2 hosts was acceptable, 2 hosts exceeded the minimum threshold value, and 1 host exceeded the medium threshold, then “medium” would be the most severe alert generated.

### 8.6.1 Statistics Server Checks

The statistics server performs regular checks and issues an alert when an alert condition is fulfilled.

The following table describes the individual checks that the statistics server performs, what actions you can take when an alert is issued for a particular check, and where you can find additional information.

<table>
<thead>
<tr>
<th>Check ID</th>
<th>Check Description</th>
<th>What to Do If an Alert Is Issued</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This check determines what percentage of the physical memory available on the host is currently used. All processes consuming memory are considered, that is, the memory used by the activities of Linux and all other programs on the host is also included.</td>
<td>Investigate memory usage. Which services are consuming a lot of memory? Are there any underlying performance-related issues, for example, high-load operations in progress, suboptimal SQL processing? You can investigate memory usage in the Administration editor on the Overview tab and the Landscape tab. The Performance tab is a good starting point for performance analysis.</td>
</tr>
</tbody>
</table>

---

Note: The thresholds configured for this check determine the color of the bar displaying memory usage values.
<table>
<thead>
<tr>
<th>Check ID</th>
<th>Check Description</th>
<th>What to Do If an Alert Is Issued</th>
</tr>
</thead>
</table>
| 2        | This check determines what percentage of space on all disks containing data, log, and trace files is currently used. This includes space used by non-SAP HANA files. To ensure that the database can always be restored to its most recent committed state and to avoid a disk-full event, there must always be enough space on disk for data and log files. **Note:** The thresholds configured for this check determine the color of the bar displaying usage values on the Overview screen of the Administration editor. | Investigate disk space usage. Which services are consuming a lot of disk space? Can space be freed up, for example, through volume shrinkage, log file deletion? Is it necessary to add additional disk space? You can investigate disk space usage in the Administration editor on the Overview tab and the Volumes tab. For more information, see:  
- Monitoring Overall System Status [page 62]  
- Monitoring System Status and Resource Usage [page 65] |
| 3        | This check determines whether or not any of the database’s services (nameserver, indexserver, preprocessor, and so on) are currently inactive. For example, if a host in a distributed system failed and a standby host took over, the services of the failed host remain inactive even after the host is reachable again. In this case, you need to restart the services manually. **Note:** The thresholds configured for this check determine the color of the bar displaying usage values on the Overview screen of the Administration editor. | Investigate why services are inactive, for example, by checking the service’s trace files. You can restart inactive services in the Administration editor on the Landscape tab. For more information, see:  
- Stopping and Starting Database Services [page 30]  
- Working with Diagnosis Files [page 220] |
| 4        | This check determines whether or not any of the database’s services (nameserver, indexserver, preprocessor, and so on) have been restarted since the last time the check was performed. | Investigate why the service had to restarted, for example, by checking the service’s trace files. For more information, see:  
- Stopping and Starting Database Services [page 30]  
- Working with Diagnosis Files [page 220] |
<p>| 5        | This check determines the percentage CPU idle time on the host and therefore whether or not CPU resources are running low. <strong>Note:</strong> The thresholds configured for this check determine the color of the bar displaying usage values on the Overview screen of the Administration editor. | Investigate CPU usage. Which services are consuming a lot of CPU resources? You can investigate CPU usage in the Administration editor on the Overview tab and the Landscape tab. |</p>
<table>
<thead>
<tr>
<th>Check ID</th>
<th>Check Description</th>
<th>What to Do If an Alert Is Issued</th>
</tr>
</thead>
</table>
| 10       | This check determines whether or not the parameter active in the mergedog section(s) of system configuration files is **yes**. mergedog is the system process that periodically checks column tables to determine whether or not a delta merge operation needs to be executed. | Set the the parameter active in the mergedog section(s) of configuration files to **yes**. For more information, see:  
  ● **Changing System Properties** [page 32]  
  ● **The Delta Merge Operation** [page 107]                                                                                                           |
| 12       | This check determines what percentage of allocated shared memory is being used by the nameserver process on a host.                                                                                                                                                   | Investigate the memory consumption of the process.  
In the Administration editor, refer to the **Landscape** tab. For more information, see:  
  ● **Monitoring System Status and Resource Usage** [page 65]  
  ● **Monitoring Memory Usage** [page 92]                                                                                                                  |
| 16       | This check determines whether the `lock_waittimeout` parameter in the transaction section of the `indexserver.ini` file is between 100,000 and 500,000.                                                                 | Change the parameter accordingly. For more information, see **Changing System Properties** [page 32]                                                                                                        |
| 17       | This check determines the number of records in non-partitioned column tables. A non-partitioned table cannot store more than 2,000,000,000 (2 billion) rows. It is possible to overcome this limit by distributing the table across several partitions. | Consider partitioning the table. For more information, see **Table Partitioning in the SAP HANA Database** [page 134]                                                                                       |
| 19       | This check determines the number of records in the delta storage of column tables. It is important that the delta storage of a table does not become excessively large and is regularly merged with the main storage. The delta merge operation is triggered in different ways depending on your system configuration. | Investigate the delta merge history to find out when the last delta merge operation was supposed to happen and why it did not. Consider merging the table delta manually. For more information, see:  
  ● **Monitoring Delta Merge History** [page 114]  
  ● **The Delta Merge Operation** [page 107]                                                                                                              |
<table>
<thead>
<tr>
<th>Check ID</th>
<th>Check Description</th>
<th>What to Do If an Alert Is Issued</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>This check determines the growth rate of non-partitioned columns tables.</td>
<td>If table growth is significant, consider partitioning the table. For more information, see <em>Table Partitioning in the SAP HANA Database</em> [page 134]</td>
</tr>
<tr>
<td>21</td>
<td>This check determines whether or not there are any internal database problems.</td>
<td>Resolve the event and mark it as resolved by executing the following SQL statement: <code>ALTER SYSTEM SET EVENT HANDLED &lt;::event&gt;</code>.</td>
</tr>
<tr>
<td>22</td>
<td>This check determines whether or not there have been any alerts since the last check and if so, sends a summary e-mail to specified recipients.</td>
<td>Investigate the alerts. For more information about configuring e-mail recipients, see <em>Configuring E-Mail Notifications for Alerts</em> [page 82]</td>
</tr>
<tr>
<td>23</td>
<td>This check determines whether or not there have been any medium and high priority alerts since the last check and if so, sends a summary e-mail to specified recipients.</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>This check determines whether or not there have been any high priority alerts since the last check and if so, sends a summary e-mail to specified recipients.</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>This check determines what percentage of the maximum number of permitted SQL connections is being used. The maximum number of permitted connections is configured in the session section of the <code>indexserver.ini</code> file.</td>
<td>Investigate why the maximum number of connections has been reached. You can cancel sessions on the Performance Sessions tab. For more information, refer to the monitoring view <code>M_CONNECTIONS</code>.</td>
</tr>
<tr>
<td>26</td>
<td>This check determines whether or not there are any unassigned volumes.</td>
<td>Investigate why the volume is unassigned. For more information, see:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <em>About Persistent Data Storage in the SAP HANA Database</em> [page 90]  • <em>Monitoring Disk Space</em> [page 88]</td>
</tr>
<tr>
<td>27</td>
<td>This check determines the number of records in column table partitions. A table partition cannot store more than 2,000,000,000 (2 billion) rows.</td>
<td>Consider re-splitting the table. For more information, see <em>Table Partitioning in the SAP HANA Database</em> [page 134]</td>
</tr>
<tr>
<td>28</td>
<td>This check determines how long ago the last savepoint was defined, that is, how long ago a</td>
<td>Investigate why there was a delay defining the last savepoint and consider triggering the</td>
</tr>
<tr>
<td>Check ID</td>
<td>Check Description</td>
<td>What to Do If an Alert Is Issued</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td></td>
<td>complete, consistent image of the database was persisted to disk. The frequency at which savepoints are defined can be configured in the persistence section of the global.ini file (every 5 minutes by default). Savepoints are also triggered automatically by a number of other operations such as data backup, and database shutdown and restart.</td>
<td>operation manually. To do so execute SQL statement <code>ALTER SYSTEM SAVEPOINT</code>.</td>
</tr>
</tbody>
</table>
| 29       | This check determines the size of the delta storage of column tables. It is important that the delta storage of a table does not become excessively large and is regularly merged with the main storage. The delta merge operation is triggered in different ways depending on your system configuration. | Investigate the delta merge history to find out when the last delta merge operation was supposed to happen and why it did not. Consider merging the table delta manually. For more information, see:  
  - Monitoring Delta Merge History [page 114]  
  - Monitoring Delta Merge Processing in Column Store Tables  
  - Performing a Manual Delta Merge Operation [page 115] |
| 30       | This check determines whether or not the disks to which data and log files are written are full. A disk-full event causes your database to stop and must be resolved. | Resolve the disk-full event. For more information, see:  
  - Resolving Disk-Full Events [page 92]  
  - About Persistent Data Storage in the SAP HANA Database [page 90] |
<p>| 31       | This check determines how many days until your license expires. | Obtain a valid license and install it. For more information, see Installing Permanent Licenses [page 40] |
| 32       | This check determines whether or not the database is running in log mode OVERWRITE. Log mode OVERWRITE does not support point-in-recovery (only recovery to a data backup) and is not recommended for productive systems. | If you need point-in-time recovery, reconfigure your system to log mode NORMAL. For more information, see Log Backup Options [page 166] |
| 33       | This check determines whether or not the database is running in log mode LEGACY. Log mode LEGACY is not recommended for productive systems. If a point-in-time recoverable system is needed the database must run in log mode NORMAL. | If you need point-in-time recovery, reconfigure your system to log mode NORMAL. For more information, see Log Backup Options [page 166] |</p>
<table>
<thead>
<tr>
<th>Check ID</th>
<th>Check Description</th>
<th>What to Do If an Alert Is Issued</th>
</tr>
</thead>
</table>
| 34       | This check determines whether or not all volumes are available so that a backup can be performed. | Investigate why the volume is not available, for example, is the associated service active? For more information, see:  
* Stopping and Starting Database Services [page 30]  
* About Persistent Data Storage in the SAP HANA Database [page 90] |
| 35       | This check determines whether or not a data backup exists.                       | To make your database recoverable, perform a data backup. For more information, see *Backing Up the SAP HANA Database* [page 176]                                                                                     |
| 36       | This check determines whether or not the most recent data backup was successful. | Investigate why the last data backup was not successful, resolve the problem, and perform a new data backup as soon as possible. For more information, see *Backing Up the SAP HANA Database* [page 176] |
| 37       | This check determines the age of the most recent data backup.                    | Perform a data backup as soon as possible. For more information, see *Backing Up the SAP HANA Database* [page 176]                                                                                     |
| 38       | This check determines whether or not the most recent log backups were successful. | Investigate why the log backup was not successful and resolve the problem. For more information, see *About SAP HANA Database Backup* [page 162]                                                                 |
| 39       | This check determines whether or not there are any SQL statements that have been running for a long time. | The table HOST_LONG_RUNNING_STATEMENTS (_SYS_STATISTICS) provides you with more detailed information about individual statements.                                                                                     |
| 40       | This check determines what percentage of the effective memory allocation limit is being consumed by a column table as a whole (that is, the cumulative size of all of a table’s columns and internal structures). | Consider partitioning or further partitioning the table.  
* Memory Management in the Column Store [page 104]  
* Memory Usage of Column and Row Tables [page 95]  
* Table Partitioning in the SAP HANA Database [page 134] |
<p>| 41       | This check determines whether or not there is a problem with the activation of an in-memory DataStore object. | For detailed information, refer to table GLOBAL_DEC_EXTRACTOR_STATUS (_SYS_STATISTICS).                                                                                                                                  |</p>
<table>
<thead>
<tr>
<th>Check ID</th>
<th>Check Description</th>
<th>What to Do If an Alert Is Issued</th>
</tr>
</thead>
<tbody>
<tr>
<td>42</td>
<td>This check finds cursors that have been open for more than the specified threshold values.</td>
<td>Close the cursor in the application, or kill the connection by executing the SQL statement <code>ALTER SYSTEM DISCONNECT SESSION &lt;LOGICAL_CONNECTION_ID&gt;</code>. For more information, see the table <code>HOST_LONG_IDLE_CURSOR (_SYS_STATISTICS)</code></td>
</tr>
</tbody>
</table>
| 43       | This check determines what percentage of its effective allocation limit a process is using. | Investigate why the process is consuming a lot of memory. You can monitor the memory usage of services in the Administration editor on the `Landscape` tab. The `Performance` tab is a good starting point for analyzing underlying performance-related issues. For more information, see:  
  - Monitoring System Status and Resource Usage [page 65]  
  - Monitoring System Performance [page 84] |
| 44       | This check determines what percentage of licensed memory is used. | Increase the licensed amount of main memory. For more information, see Installing Permanent Licenses [page 40] |
| 45       | This check determines what percentage of the effective memory allocation limit is being consumed the main storage of a column table. | Consider partitioning or further partitioning the table. For more information, see:  
  - Memory Management in the Column Store [page 104]  
  - Memory Usage of Column and Row Tables [page 95]  
  - Table Partitioning in the SAP HANA Database [page 134] |
| 46       | This check determines whether or not new runtime dump files (RTE) have been generated in the trace directory of the system. These contain information about, for example, build, loaded modules, running threads, CPU, and so on. SAP Support may want to examine these files for problem analysis. | Check the contents of the RTE files. For more information, see Working with Diagnosis Files [page 220] |
### Check Description

<table>
<thead>
<tr>
<th>Check ID</th>
<th>Check Description</th>
<th>What to Do If an Alert Is Issued</th>
</tr>
</thead>
<tbody>
<tr>
<td>47</td>
<td>This check determines whether or not there are any long-running serializable transactions.</td>
<td>Close the serializable transaction in the application or kill the connection by executing the following SQL statement: <code>ALTER SYSTEM DISCONNECT SESSION &lt;LOGICAL_CONNECTION_ID&gt;</code>. For more information, see the table HOST_LONG_SERIALIZABLE_TRANSACTION (_SYS_STATISTICS).</td>
</tr>
<tr>
<td>48</td>
<td>This check determines whether or not there are any uncommitted transactions. Such transactions can impact the performance of the database.</td>
<td>Close the uncommitted transaction in the application or kill the connection by executing the following SQL statement: <code>ALTER SYSTEM DISCONNECT SESSION &lt;LOGICAL_CONNECTION_ID&gt;</code>. For more information, see the table HOST_LONG_UNCOMMITTED_WRITE_TRANS (_SYS_STATISTICS).</td>
</tr>
<tr>
<td>49</td>
<td>This check determines whether or not there are any blocked transactions.</td>
<td>Review the blocking and blocked transactions, for example, on the Performance &gt; Threads and if appropriate cancel one of them. For more information, see Monitoring System Performance [page 84]</td>
</tr>
<tr>
<td>50</td>
<td>This check determines the number of diagnosis files written by the system. A unusually large number of files can indicate a problem with the database (for example, problem with trace file rotation or a high number of crashes).</td>
<td>Check the diagnosis files. For more information, see Working with Diagnosis Files [page 220]</td>
</tr>
<tr>
<td>51</td>
<td>This check determines the size of individual diagnosis files. An unusually large file can indicate a problem with the database.</td>
<td>Check the diagnosis file in question. For more information, see Working with Diagnosis Files [page 220]</td>
</tr>
</tbody>
</table>

### Related Links

* SAP HANA System Tables and Monitoring Views Reference

### 8.6.2 Configuring E-Mail Notifications for Alerts

You can configure the statistics server in such a way that you receive an e-mail when an alert condition for all or specific checks is fulfilled.

To be able to configure check settings, you must have the system privilege INIFILE ADMIN.

1. In the Administration editor, choose the Alerts tab.

2. From the tab toolbar, choose the (Configure Check Settings) button. The Configure Check Settings dialog box appears.
3. Enter the following information:
   ○ Sender's e-mail address
     E-mail address that is entered as the email's sender
   ○ SMTP server
     The mailserver that the statistics server sends the e-mails to
   ○ Optional: SMTP Port
     The default SMTP port is 25. If the configured mailserver uses a different port, enter it.

4. Optional: Specify the recipient(s) to whom you want an e-mail notification to be sent when an alert is generated for any check.
   To do so, choose Modify Recipients and add the e-mail addresses of the users.

   **Note:** You can omit this step and only configure e-mail notification for specific checks (next step).

5. Specify the recipients to whom you want an e-mail notification to be sent when an alert is generated for a specific check or checks.
   a) Choose Recipients Configuration for Specific Checks.
   b) Select the checks for which you want to configure e-mail notification and then choose Add Recipients to add the e-mail addresses of the users to be notified.

6. Choose OK to save the configuration.

The specified recipients are notified by e-mail when the statistics server generates an alert for the relevant checks.

8.6.3 Configuring Check Thresholds

For some checks performed by the statistics server, you can configure when an alert is issued, that is the alert condition. A check can have a low, medium, and high priority threshold.

To be able to configure check thresholds, you must have the system privilege INIFILE ADMIN.

1. In the Administration editor, choose the Alerts tab.

2. From the tab toolbar, choose the (Configure Check Settings) button. The Configure Check Settings dialog box appears.

3. Choose the Configure Check Thresholds tab.

4. Choose the check that you want to change and enter the threshold values.
   The threshold value and unit depend on what is being measured. For example, for check 2 (disk usage), you could enter 90, 95 and 100 as the thresholds, where 90, 95 and 100 represent the percentage of disk space used.

5. Choose OK when you have finished configuring the check thresholds.

Alerts are generated when the configured threshold values are measured by the statistics server. The color of the bar views on the Overview tab may also change when certain thresholds are changed. For example, you change the disk space threshold from 90, 95 and 100, to 85, 90 and 95. If the disk is at 95% usage, then the bar view would change from yellow to red.
8.6.4 Configuring Start Times of Periodic Checks

Some statistics server checks are performed every 6 or 24 hours. For example, the age of the last data backup is checked every 24 hours. You can configure the start times for these checks.

To be able to configure the start times of periodic checks, you must have the system privilege INIFILE ADMIN.

1. In the Administration editor, choose the Alerts tab.
2. From the tab toolbar, choose the Configure Check Settings button. The Configure Check Settings dialog box appears.
3. Choose the Configure Start Time on Check Intervals tab. This tab has two sections, one for setting the start time of checks performed every 6 hours and one for setting the start time of checks performed every 24 hours. You can also see which checks are performed at these intervals.
4. Set the start time for both periodic check types.
5. Choose OK.

The start time for the checks is changed.

8.7 Monitoring System Performance

In addition to the general information about the overall system performance that you can see in the System Monitor and on the Overview tab of the Administration editor, you can monitor more detailed aspects of system performance on the Performance tab.

The following tables provides an overview of the information available on the Performance tab.

Note: The available filters and column configuration options allow you to restrict the amount of information displayed. You can also configure the information that is displayed in the various table views (that is, visible columns) in the Table Viewer. To open the Table Viewer, choose Configure Viewer in the toolbar on the top-right of the screen.

<table>
<thead>
<tr>
<th>Sub-Tab</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threads</td>
<td>By default, the Threads sub-tab shows you a list of all currently active threads in your landscape. It may be useful to see, for example, how long a thread is running, if a thread is blocked for an inexplicable length of time, and so on. A blocked thread is indicated by a warning icon (⚠️) in the Status column, and you can see detailed information about the blocking situation by hovering the cursor over this icon. The following features are also available on the Threads tab:</td>
</tr>
<tr>
<td>Sub-Tab</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| ● You can end the operation of a specific thread by right-clicking the thread and choosing **Cancel Operations**.  
● You can view the call stack for a specific thread. To do so, select the **Create call stacks** checkbox, refresh the page, and then select the thread in question. |
| Sessions | The **Sessions** sub-tab allows you to monitor all the sessions in your landscape. For example, you can identify inactive sessions, investigate blocking situations, and view memory consumption by session.  
You can cancel a session by right-clicking the session and choosing **Cancel Session...** |
| SQL plan cache | The **SQL plan cache** is useful for observing overall SQL performance as it provides statistics on compiled queries. Here, you can get insight into frequently executed queries and slow queries with a view to finding potential candidates for optimization.  
The following information may be useful:  
● Dominant statements (TOTAL_EXECUTION_TIME)  
● Long-running statements (AVG_EXECUTION_TIME)  
● Frequently-executed plans (EXECUTION_COUNT)  
● Number of records returned (TOTAL_RESULT_RECORD_COUNT)  
● Statements with high lock contention (TOTAL_LOCK_WAIT_COUNT)  

1. **Note:** The collection of SQL plan cache statistics is enabled by default, but you can disable it on the **SQL Plan Cache** tab by choosing **Configure**.  
To help you understand and analyze the execution plan of an SQL statement further, you can generate a graphical view of its plan by right-clicking the statement and choosing **Visualize Plan**. |
| Expensive statements | Expensive statements are individual SQL queries whose execution time was above a configured threshold. Expensive statements may reduce the performance of the database.  
The following information may be useful:  
● When the query started (START_TIME)  
● How long the query took (DURATION_MICROSEC)  
● Name(s) of the objects accessed (OBJECT_NAME)  
● The SQL statement (STATEMENT_STRING)  

1. **Note:** The expensive statements trace is deactivated by default. You can activate it either here on the **Expensive Statements Trace** sub-tab, or on the **Trace Configuration** tab. You can also configure the threshold value here. The default threshold value is 1000000 microseconds, that is, 1 second.  
To help you understand and analyze the execution plan of an expensive statement further, you can generate a graphical view of its plan by right-clicking the statement and choosing **Visualize Plan**. |
| Job progress of long-running operations | Certain operations typically run for a long time and may consume a considerable amount of resources, for example, delta merge operations, data compression, and delta log replays. By monitoring the progress of these long-running transactions, you can |
### System Tables and Monitoring Views

The SAP HANA database provides many system tables and monitoring views that contain important information about the database. Much of the information in these tables and views is available on the various tab pages of the Administration editor. However, it can be necessary to examine the data directly as part of more detailed performance analysis.

**Note:** Many monitoring views are available in two versions – one that shows the data gathered since the service in question was last started, and one that shows the data gathered since the time the table was last reset. For example, the monitoring view M_VOLUME_IO_STATISTICS shows the total read size and the total write size for each volume since the service in question was last started. The SQL command

```
ALTER SYSTEM RESET MONITORING VIEW SYS.M_VOLUME_IO_STATISTICS_RESET
```

initializes the statistics shown by this view. The monitoring view M_VOLUME_IO_STATISTICS_RESET now shows the statistics since the reset time.

You can access tables and views in the following ways:

- **On the System Information tab of the Administration editor**

  Several pre-defined SQL SELECT statements on system tables and monitoring views are available here as tables. They provide you with easy access to important system information. Double-clicking a table opens it for data analysis.

- **Open directly in the table editor**

  For example, you can search for a particular monitoring view (they all start with M_*) by right-clicking your system in the Navigator and choosing Find Table.
Historical Performance Monitoring

The statistics server is a valuable resource for performance analysis. It periodically gathers data from system tables and monitoring views and stores it in tables in the schema _SYS_STATISTICS. You can use these tables to analyze system behavior over time. You can open statistics server tables directly in the table editor.

SQL Execution

You can investigate the performance of SQL statement execution with the help of the following functions of the SQL console:

- Execution plan explanation
- Execution plan visualization

Traces and Profilers

You can activate various traces and profilers (SQL trace, performance trace, kernel profiler) to help you analyze performance issues.

Related Links

SAP HANA System Tables and Monitoring Views Reference

Opening Tables in the Table Editor [page 120]
Some monitoring and problem analysis may require you to examine individual tables, for example, the many system tables and monitoring views provided by the SAP HANA database. You can open tables and views directly in the table editor. Different viewing options are available depending on what you want to do.

Analyzing SQL Execution with the Plan Explanation [page 209]
You can generate a plan explanation for any SQL statement in the SQL console. You can use this to evaluate the execution plan that the SAP HANA database follows to execute an SQL statement.

Analyzing SQL Execution with the Plan Visualizer [page 209]
To help you understand and analyze the execution plan of an SQL statement, you can generate a graphical view of the plan.

SQL Trace [page 222]
The SQL trace collects information about all executed SQL statements and saves it in a trace file for further analysis.

Performance Trace [page 223]
The performance trace is a performance tracing tool built into the SAP HANA database. It records performance indicators for individual query processing steps in the database kernel.

Kernel Profiler [page 224]
The kernel profiler is a sampling profiler built into the SAP HANA database. It collects, for example, information about frequent and/or expensive execution paths during query processing.
8.8 Monitoring Disk Space

To ensure that the database can always be restored to its most recent committed state, you must ensure that there is enough space on disk for data and log volumes. You can monitor disk usage, volume size, and other disk activity statistics on the Volumes tab of the Administration editor.

There are two views available on the Volumes tab for monitoring the size of volumes on disk: service and storage type (that is, data, log, and trace).

**Note:** Although trace files are not stored in volumes, they are displayed on the Volumes tab in the Storage view as they consume disk space and therefore need to be monitored.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service/Volume</td>
<td>The service host and internal port</td>
</tr>
<tr>
<td></td>
<td>You can expand the host/port to can see the storage area for data and log.</td>
</tr>
<tr>
<td>Service</td>
<td>The name of the service that has a data and log volume</td>
</tr>
<tr>
<td>Total Volume Size [MB]</td>
<td>Total size of the service's data and log volumes</td>
</tr>
<tr>
<td></td>
<td>If you expand the host/port, you can see the size of each volume.</td>
</tr>
<tr>
<td>Data Volume Size [MB]</td>
<td>Current size of the service’s data volume</td>
</tr>
<tr>
<td>Log Volume Size [MB]</td>
<td>Current size of the service’s log volume</td>
</tr>
<tr>
<td>Path</td>
<td>Location of the service’s data and log files in the file system</td>
</tr>
<tr>
<td>Storage Device ID</td>
<td>ID of the device on which the data and log files are stored</td>
</tr>
<tr>
<td></td>
<td>This can be useful for checking whether or not data and log files are on the same device.</td>
</tr>
<tr>
<td>Total Disk Size [MB]</td>
<td>Total size of the host’s hard disk</td>
</tr>
<tr>
<td>Used Disk Size [MB]</td>
<td>Amount of disk space used on the host’s hard disk as a whole</td>
</tr>
<tr>
<td>Available Disk Size [%]</td>
<td>Available disk space on the host’s hard disk</td>
</tr>
</tbody>
</table>

The information shown when you select the Storage view is the same as above. It is simply displayed according to storage type not service. Details of trace files stored on disk are also available in this view.

When you select a row in either view, detailed information about the volume(s) is displayed in the lower part of the screen. In addition to size and usage information, statistics relating to the performance of read/write operations to disk are also available.

**Note:** Detailed information about nameserver volumes is currently not available.
### Table 11: Volume Details View

<table>
<thead>
<tr>
<th>Tab Page</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Files**                      | This tab page displays the file name and type. It also shows the size of the file and how much of it is currently in use, both in MB and as a percentage of its total size. The relevance of used size depends on the file type as follows:  
  - **Data files**  
    Used size is the amount of data in the file. As the size of the file is automatically increased with the payload but not automatically decreased, used size and total size may be different.  
  - **Log segment files**  
    Used size equals total size. When a file is full, log entries are written to the next log segment file available. The log segment file’s state indicates its availability for reuse. For more information, see the monitoring view M_LOG_SEGMENTS.  
  - **Trace files**  
    Used size is zero for unused trace files and equals total size for used trace files.                                                                                                                                                                                                                     |
| **Volume I/O Statistics**      | This tab page shows aggregated I/O statistics for the volume since the service was started, for example, number of read/write requests, data throughput, total I/O time, and speed (MB/s). These figures can be useful when analyzing performance problems.  
  For more information about the meaning of the individual fields, see the monitoring view M_VOLUME_IO_STATISTICS.                                                                                                                                                                               |
| **Data Volume Superblock Statistics** | This tab page displays aggregated statistics on the data volume’s superblocks since the service was started.  
  Superblocks are partitions of the data volume that contain pages of the same page size class.  
  For more information about the meaning of the individual fields, see monitoring view M_DATA_VOLUME_SUPERBLOCK_STATISTICS.                                                                                     |
| **Data Volume Page Statistics** | This tab page displays statistics on the data volume’s pages (or blocks) broken down according to page size class. You can analyze how many superblocks are used for the specific size class and also how many pages/blocks are used. The fill ratio enables you to decide whether or not it makes sense to reorganize and release unnecessary superblocks, in other words, shrink the data volume.  
  For more information about the meaning of the individual fields, see monitoring view M_DATA_VOLUME_PAGE_STATISTICS.                                                                                                          |

**Related Links**

- **Diagnosis Files** [page 219]  
  Diagnosis files include log and trace files, as well as a mixture of other diagnosis, error, and information files. In the event of problems with the SAP HANA database, you can check these diagnosis files for errors.  

- **SAP HANA Database - System Tables and Monitoring Views**
8.8.1 About Persistent Data Storage in the SAP HANA Database

To ensure that the database can always be restored to its most recent committed state, changes to data in the database is periodically copied to disk. Logs containing data changes and certain transaction events are also saved regularly to disk. The data and logs of a system are stored in volumes.

During the savepoint operation, the SAP HANA database flushes all changed data from memory to the data volumes. The data belonging to a savepoint represents a consistent state of the data on disk and remains so until the next savepoint operation has completed. Redo log entries are written to the log volumes for all changes to persistent data. In the event of a database restart (for example, after a crash), the data from the last completed savepoint can be read from the data volumes, and the redo log entries written to the log volumes since the last savepoint can be replayed.

You must always ensure that there is enough space on the disk to save data and logs. Otherwise, a disk-full event will occur and your database will stop working.

Directory Hierarchy for Data and Log Storage

During the installation process, the following default directories are created as the storage locations for data and log volumes respectively:

- `/usr/sap/<SID>/global/hdb/data`
- `/usr/sap/<SID>/global/hdb/log`

Note: These default directories are defined in the parameters `basepath_datavolumes` and `basepath_logvolumes` in the `persistence` section of the `global.ini` file.

These directories contain a separate sub-directory, or storage partition, for each host in the system. These are named `mnt00001`, `mnt00002`, `mnt00003` and so on, by default. Each host storage partition contains a sub-directory for every database service that persists data. These sub-directories represent the actual volumes. They are named `hdb00001`, `hdb00002`, `hdb00003`, and so on by default. The services that persist data and therefore have volumes are as follows:

<table>
<thead>
<tr>
<th>Service</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>namesever</td>
<td>Only the nameserver service on the active master host persists data. Slave nameserver hosts communicate with the master, but do not persist data.</td>
</tr>
<tr>
<td>indexserver</td>
<td>The indexserver service on all hosts except standby hosts persists data.</td>
</tr>
<tr>
<td>statisticsserver</td>
<td>The statisticsserver service runs only on one host and persists data on this host.</td>
</tr>
<tr>
<td>xsengine (if running)</td>
<td>The xsengine service persists data on any host on which it is running.</td>
</tr>
</tbody>
</table>

The following figure illustrates the storage hierarchy described above using the example of data storage.

*Figure 3: Directory Hierarchy for Persistent Data Storage*
Data and Log Volumes

Each data volume contains one file (datavolume_0000.dat) in which data is organized into pages, ranging in size from 4KB to 16MB (page size class). Data is written to and loaded from the data volume page-wise. Over time, pages are created, changed, overwritten, and deleted. The size of the data file is automatically increased as more space is required. However, it is not automatically decreased when less space is required. This means that at any given time, the actual payload of a data volume (that is the cumulative size of the pages currently in use) may be less than its total size. This is not necessarily significant — it simply means that the amount of data in the file is currently less than at some point in the past (for example, after a large data load). If a data volume has a considerable amount of free space, it might be appropriate to shrink the data volume. However, a data file that is excessively large for its typical payload can also indicate a more serious problem with the database. SAP support can help you analyze your situation.

Each log volume contains the file logsegment_<partition_ID>directory.dat and one or more log segment files (logsegment_<partition>_<segment_number>.dat). Currently only one log partition is supported for each service, so the default file names are logsegment_000_directory.dat and logsegment_000_00000000.dat, logsegment_000_00000001.dat, logsegment_000_00000002.dat and so on. Log segment files are cyclically overwritten depending on the log mode. The log mode determines how logs are backed up. Log volumes only grow if there are no more segment files available for overwriting. Log segment files that are available for overwriting have been backed up and are not required for a database restart. If necessary you can remove these files to free up space in your file system by executing the SQL statement ALTER

SAP HANA Administration Guide
Monitoring SAP HANA Systems

© 2012 SAP AG. All rights reserved.
PUBLIC
91
SYSTEM RECLAIM LOG. Note that new log segment files will need to be created later and this will affect performance.

Note: Do not remove either data files or log files using operating system tools as this will corrupt the database.

Related Links
Data Volume Encryption [page 152]
To ensure that your database can always be restored to its most recent committed state, all data in the database is periodically copied to disk. You can ensure the privacy of data on disk by enabling persistence encryption.

8.8.2 Resolving Disk-Full Events
When the disks on which the database data and log volumes are located run full, the database is suspended, an internal event is triggered, and an alert is generated. A disk-full event must be resolved before the database can resume.

To be able to resolve a disk-full event, you must have system privilege MONITOR ADMIN.

If a disk-full event occurs, in addition to the size and usage information contained on the Overview tab of the Administration editor, a Disk Full Event field is displayed.

Note: You can also view the alert information on the Alerts tab.

1. On the Volumes tab, set the Show filter to Storage.
2. Check whether it is the database that is using all the space or additional files.
3. If additional files are stored on the SAP HANA storage system, remove any that are not needed, or add additional storage space.
4. On the Overview tab, choose the Disk Full Events link.
5. Mark the event as handled.

The database resumes.

8.9 Monitoring Memory Usage
Memory is a fundamental resource of the SAP HANA database. Understanding how the SAP HANA database requests, uses, and manages this resource is crucial to the understanding of SAP HANA.

The SAP HANA database provides many monitoring views and memory indicators that allow you to monitor and understand SAP HANA memory usage. The most important indicators are used memory and peak used memory. Since SAP HANA contains its own memory manager and memory pool, external indicators, such as the size of resident memory at host level and the size of virtual and resident memory at process level can be misleading when you are estimating the real memory requirements of an SAP HANA deployment.

You can view the most important indicators of memory usage on the Overview tab of the Administration editor.

For more information about memory consumption with regards to SAP HANA licenses, see SAP Note 1704499 (License Memory Audit).
When you open the Administration editor for a particular SAP HANA system, the Overview tab provides you with a summary of the overall status of the system, as well as an overview of resource usage.

**SAP Note 1704499**

### 8.9.1 What Is Memory Used For?

The total amount of memory used by SAP HANA is referred to as used memory and includes program code and stack, all data and system tables, and the memory required for temporary computations.

In the Linux operating environment, memory is allocated for the program code (sometimes called the text), the program stack, and data. Most of the data memory, called the heap, is under program control.

As an in-memory database, it is critical for SAP HANA to manage and track its own consumption of memory carefully. For this purpose, the SAP HANA database pre-allocates and manages its own data memory pool. The memory pool is used for storing in-memory tables, for thread stacks, as well as for temporary computations, intermediate results, and other data structures.

SAP HANA’s utilization of memory thus includes its program code (exclusive and shared), the program stack, and the memory pool, which includes all data tables (row and column), system tables, and created tables. At any given time, parts of the pool are in use for temporary computations. The total amount of memory in use is referred to as used memory. This is the most precise indicator of the amount of memory that the SAP HANA database uses.

The following figure shows used memory, consisting of code, stack, and table data:

![Figure 4: SAP HANA Used Memory](image)

Since the code and program stack size are less than 3 GB, almost all of used memory is used for storing tables and for computations.

### Memory Sizing

Memory sizing is the process of estimating in advance the amount of memory that will be required to run a certain workload on an SAP HANA database. To understand memory sizing, the following questions need to be answered:

- What is the size of the data tables that will be stored in the SAP HANA database?
You may be able to estimate this based on the size of your existing data, but unless you precisely know the compression ratio of the existing data and the anticipated growth factor, this estimate may not be accurate.

- What is the expected compression ratio that SAP HANA will apply to these tables?

The column store of the SAP HANA database automatically uses a combination of various advanced compression algorithms (dictionary, LRE, sparse, and so on) to compress each table column separately. The achieved compression ratio depends on many factors, such as the nature of the data, its organization and data types, the presence of repeated values, the number of indexes (SAP HANA requires fewer indexes), and so on.

- How much extra working memory will be required for temporary computations?

The amount of extra memory will depend on the size of the tables (larger tables will create larger intermediate result tables in operations such as joins), but even more on the expected workload in terms of the concurrency and complexity of analytical queries (each concurrent query needs its own workspace).

SAP Notes 1514966, 1637145, and 1736976 provide additional tools and information to help you size the required amount of memory. However, the most accurate method is to import several representative tables into an SAP HANA system, measure the memory requirements, and extrapolate from the results.

Related Links
SAP Note 1514966
SAP Note 1637145
SAP Note 1736976

8.9.2 SAP HANA Used Memory and Peak Used Memory

In addition to used memory, SAP HANA has a further used memory indicator called peak used memory. This is useful for keeping track of the maximum value for used memory over time.

The total amount of memory in use by SAP HANA is referred to as its used memory. This is the most precise indicator of the amount of memory that the SAP HANA database is using at any time.

To display the current size of used memory, you can execute the following SQL statement (for example, in the SQL console of the SAP HANA studio):

```
SELECT ROUND(SUM(TOTAL_MEMORY_USED_SIZE/1024/1024/1024), 2) AS "Used Memory GB" FROM SYS.M_SERVICE_MEMORY;
```

Peak Used Memory

The value for used memory is a current measurement. Ultimately, it is more important to understand the behavior of used memory over time and under peak loads. For this purpose, SAP HANA has a special used memory indicator called peak used memory. This is useful for keeping track of the maximum value for used memory over time.
You can read peak used memory on the Overview screen of the Administration editor, or by executing the following SQL statement:

```
SELECT ROUND(SUM("M")/1024/1024/1024,2) as "Peak Used Memory GB" FROM (SELECT SUM(CODE_SIZE+SHARED_MEMORY_ALLOCATED_SIZE) AS "M" FROM SYS.M_SERVICE_MEMORY UNION SELECT SUM(INCLUSIVE_PEAK_ALLOCATION_SIZE) AS "M" FROM M_HEAP_MEMORY_RESET WHERE DEPTH = 0)
```

You can also reset peak used memory. This can be useful if you want to establish the impact of a certain workload on memory usage. So for example, you can reset peak used memory, run the workload, and then examine the new peak used memory value.

You can reset peak used memory on the Landscape Services tab by choosing Reset Memory Statistics from the context menu. Alternatively, you can execute the following statement:

```
ALTER SYSTEM RESET MONITORING VIEW SYS.M_HEAP_MEMORY_RESET
```

**Maximum Peak Used Memory**

In addition to the resettable peak used memory value, the SAP HANA database also tracks the highest-ever value of used memory since the database was started. In fact, this is probably the single most significant memory indicator that you should monitor as an overall indicator of the total amount of memory required to operate the SAP HANA database over a long period of time.

To display the maximum peak used memory, execute the following SQL statement:

```
SELECT ROUND(SUM("M")/1024/1024/1024,2) as "Max Peak Used Memory GB" FROM (SELECT SUM(CODE_SIZE+SHARED_MEMORY_ALLOCATED_SIZE) AS "M" FROM SYS.M_SERVICE_MEMORY UNION SELECT SUM(INCLUSIVE_PEAK_ALLOCATION_SIZE) AS "M" FROM M_HEAP_MEMORY WHERE DEPTH = 0)
```

**Note:** If peak used memory is never reset, it equals maximum peak used memory

### 8.9.3 Memory Usage of Column and Row Tables

A significant part of the used memory value in the SAP HANA database is the space used by data tables. Separate measurements are available for column-store tables and row-store tables.

**Column-Store Tables**

To get a high-level overview of the amount of memory used for column-store tables, you can execute the following SQL statement:

```
SELECT ROUND(SUM(MEMORY_SIZE_IN_TOTAL)/1024/1024) AS "Column Tables MB Used" FROM M_CS_TABLES
```
To get a breakdown by schema, you can execute the following statement:

```sql
SELECT SCHEMA_NAME AS "Schema", ROUND(SUM(MEMORY_SIZE_IN_TOTAL) / 1024/1024) AS "MB Used" FROM M_CS_TABLES GROUP BY SCHEMA_NAME ORDER BY "MB Used" DESC
```

Note that the SAP HANA database loads column-store tables into memory column by column only upon use. This is sometimes called “lazy loading”. This means that columns that are never used will not be loaded and memory waste is avoided. When the SAP HANA database runs out of allocatable memory, it will try to free up some memory by unloading unimportant data (such as caches) and even table columns that have not been used recently. Therefore, if it is important to measure precisely the total, or worst-case, amount of memory used for a particular table, it is important to ensure that the table is first fully loaded into memory. You can do this by executing the following SQL statement:

```sql
LOAD table_name ALL
```

The following example shows you how you can examine the amount of memory consumed by the tables in a specific schema (SYSTEM):

```sql
SELECT TABLE_NAME AS "Table", ROUND(MEMORY_SIZE_IN_TOTAL/1024/1024, 2) as "MB Used" FROM M_CS_TABLES WHERE SCHEMA_NAME = 'SYSTEM' ORDER BY "MB Used" DESC
```

If you want to drill down into columns of the table LineItem further, to view the actual size of the data, the size of data in delta storage, and the compression ratio for each of its columns, you could execute the following statement for example:

```sql
SELECT COLUMN_NAME AS "Column", LOADED, ROUND(UNCOMPRESSED_SIZE/1024/1024) AS "Uncompressed MB", ROUND(MEMORY_SIZE_IN_MAIN/1024/1024) AS "Main MB", ROUND(MEMORY_SIZE_IN_DELTA/1024/1024) AS "Delta MB", ROUND(MEMORY_SIZE_IN_TOTAL/1024/1024) AS "Total Used MB", ROUND(COMPRESSION_RATIO_IN_PERCENTAGE/100, 2) AS "Compr. Ratio" FROM M_CS_Columns WHERE TABLE_NAME = 'LineItem'
```

Note: The above statements query the system views M_CS_TABLES and M_CS_COLUMNS. These monitoring views contain much more information (such as cardinality, main storage, delta storage, and so on).

**Row-Store Tables**

Many system tables, such as statistics server tables, are in fact row store tables. To get a sense of the total amount of memory used for these row tables, you can use the following query:

```sql
SELECT ROUND(SUM(USED_FIXED_PART_SIZE + USED_VARIABLE_PART_SIZE)/1024/1024) AS "Row Tables MB Used" FROM M_RS_TABLES
```
To examine the memory consumption of the row-store tables of a particular schema, for example, the schema SYS, you can drill down by executing the following statement:

```sql
SELECT SCHEMA_NAME, TABLE_NAME, ROUND((USED_FIXED_PART_SIZE + USED_VARIABLE_PART_SIZE)/1024/1024, 2) AS "MB Used" FROM M_RS_TABLES WHERE SCHEMA_NAME = 'SYS' ORDER BY "MB Used" DESC, TABLE_NAME
```

**Related Links**

*Loading and Unloading Column Tables into and from Memory* [page 105]

Under normal circumstances, the SAP HANA database manages the loading and unloading of tables into and from memory independently – the aim being to keep all relevant data in memory. However, you can manually load and unload individual tables and table columns if necessary.

### 8.9.4 Allocated Memory Pools and Allocation Limits

The SAP HANA database, across its different processes, reserves a pool of memory before actual use. This pool of allocated memory is preallocated from the operating system over time, up to a predefined global allocation limit, and is then efficiently used as needed by the SAP HANA database code.

More memory is allocated to the pool as memory consumption increases. If the amount of memory used nears the global allocation limit, the SAP HANA database may run out of memory if it cannot free memory.

This preallocation of pool memory is the reason why Linux memory indicators (such as top and meminfo) do not accurately reflect the actual SAP HANA used memory size.

By default, the allocation limit is set to 90% of the size of the host’s physical memory. You can see what the allocation limit of the database is in the Administration editor on the *Overview* tab.

There is normally no reason to change the value of this variable, unless you purchased a license for less than the total amount of physical memory. In this case, you need to change the global allocation limit to remain in compliance with the license.

**Example:**

- You have a server with 512GB, but purchased an SAP HANA license for only 384 GB. You therefore set the `global_allocation_limit` to 393216 (384 * 1024 MB).
- You have a distributed HANA system on four hosts with 512 GB each, but purchased an SAP HANA license for only 768 GB. Set the `global_allocation_limit` to 196608 (192 * 1024 MB on each host).

In addition to the global allocation limit, each service running on the host has an allocation limit, the *service allocation limit*. Given that collectively, all services cannot consume more memory than the global allocation limit, each service has what is called an *effective allocation limit*. The effective allocation limit of a service specifies how much physical memory a service can in reality consume given the current memory consumption of other services.

You can see what the current effective allocation limit of a service is in the Administration editor on the *Landscape* tab -> *Services* tab.

**Example:** A single-host system has 100 GB physical memory. Both the global allocation limit and the individual service allocation limits are 90% (default values).

This means the following:

- Collectively, all services of the HANA database can use a maximum of 90 GB.
Individually, each service can use a maximum of 90 GB.

Therefore, if 2 services are running and the current memory pool of service 1 is 50 GB, then the effective allocation limit of service 2 is 40 GB. This is because service 1 is already using 50 GB and together they cannot exceed the global allocation limit of 90 GB.

Related Links
Setting the global_allocation_limit Parameter [page 33]

The SAP HANA database preallocates a pool of memory from the operating system over time, up to a predefined global allocation limit. You can change the default global allocation limit in the global.ini configuration file.

8.9.5 SAP HANA Memory Usage and the Operating System

Due to the way in which SAP HANA manages memory, the relationship between Linux memory indicators and SAP HANA’s own memory indicators may not correlate as expected.

From the perspective of the Linux operating system, SAP HANA is a collection of separate processes. Linux programs reserve memory for their use from the Linux operating system. The entire reserved memory footprint of a program is referred to as its virtual memory. Each Linux process has its own virtual memory, which grows when the process requests more memory from the operating system, and shrinks when the process relinquishes unused memory. You can think of virtual memory size as the memory amount that the process has requested (or allocated) from the operating system, including reservations for its code, stack, data, and memory pools under program control. SAP HANA’s virtual memory is logically shown in the following figure:

Figure 5: SAP HANA Virtual Memory

Note: SAP HANA really consists of several separate processes, so the figure above shows all SAP HANA processes combined.

Virtual, Physical, and Resident Memory

When part of the virtually allocated memory actually needs to be used, it is loaded or mapped to the real, physical memory of the host and becomes resident. Physical memory is the DRAM memory installed on the host. On
most SAP HANA hosts, it ranges from 256 gigabytes (GB) to 1 terabyte (TB). It is used to run the Linux operating system, SAP HANA, and all other programs.

Resident memory is the physical memory actually in operational use by a process. Over time, the operating system may swap out some of a process’s resident memory according to a least-recently-used algorithm to make room for other code or data. Thus, a process’s resident memory size may fluctuate independently of its virtual memory size. In a properly-sized SAP HANA appliance, there is enough physical memory, so that swapping is disabled and should not be observed.

This can be illustrated as follows:

On a typical SAP HANA appliance, the resident memory part of the operating system and all other running programs usually does not exceed 2 GB. The rest of the memory is therefore dedicated for the use of SAP HANA.

To display the size of physical memory and the resident part, you can use the following SQL command:

```sql
SELECT HOST, ROUND((USED_PHYSICAL_MEMORY + FREE_PHYSICAL_MEMORY) / 1024/1024/1024, 2) AS "Physical Memory GB", ROUND(USED_PHYSICAL_MEMORY / 1024/1024/1024, 2) AS "Resident GB" FROM PUBLIC.M_HOST_RESOURCE_UTILIZATION
```

When memory is required for table growth or for temporary computations, the SAP HANA code obtains it from the existing memory pool. When the pool cannot satisfy the request, the SAP HANA memory manager will request and reserve more memory from the operating system. At this point, the virtual memory size of SAP HANA processes grows.

Once a temporary computation completes or a table is dropped, the freed memory is returned to the memory manager, which recycles it to its pool without informing the operating system. Therefore, from SAP HANA’s perspective, the amount of used memory shrinks, but the processes’ virtual and resident memory sizes are not...
affected. This creates a situation where the used memory value may shrink to below the size of SAP HANA’s resident memory. This is normal.

The following illustration shows the relationship between physical memory, Linux virtual memory and resident memory, and SAP HANA’s allocated memory and used memory indicators. Note how changes in used memory do not affect the processes’ virtual and resident memory sizes.

**Figure 7: Relationship Between SAP HANA and Linux Memory Indicators**

![Diagram showing memory relationships](image)

**Linux Indicators**

To view the amount of physical memory on the host, and the resident part, execute the following command:

```bash
free -g | awk '/Mem:/ {print "Physical Memory: " $2 " GB."} /cache:/ {print "Resident: " $3 " GB."}''
```

This merely provides the host perspective. The Linux process indicators are more relevant. Since the operating system treats SAP HANA as a collection of processes, we need to consider both the virtual and resident part of these processes. A process’s virtual memory size is always larger than its resident size. Note also that due to the managed memory pool, both the SAP HANA virtual and resident memory sizes may appear larger than what the used memory indicator would lead you to expect. This is entirely normal.

Linux maintains high water mark (peak) indicators for the virtual and resident process sizes. In a stable system, the current virtual and resident sizes will be only slightly smaller than their respective high water marks because SAP HANA grows its pool, but does not normally relinquish unused memory. Thus, the resident size high water mark should generally track the maximum peak used memory. Very large differences may indicate that parts of the SAP HANA memory pool were freed, possibly due to insufficient physical memory.

You can display these process memory indicators for SAP HANA as follows (replace “qp4adm” in this example with the user name of the appropriate SAP HANA administrator):

```bash
cat `ps h -U qp4adm -o "/proc/%p/status" | tr -d '"' | awk '/VmSize/ {v+=\$2} /VmPeak/ {vp+=\$2} /VmRSS/ {r+=\$2} /VmHWM/ {rp+=\$2} END {printf("Virtual Size = %.2f GB (peak = %.2f), Resident size = %.2f GB (peak = %.2f)\n", v/1024/1024, vp/1024/1024, r/1024/1024, rp/1024/1024)}`
```

---

© 2012 SAP AG. All rights reserved.
Process memory reports on Linux are difficult to interpret for various reasons, such as shared-memory double counting. Thus, the above values, which represent the sum of sizes of all the HANA processes, will be slightly inaccurate, but not hugely so.
9 Managing Tables in the SAP HANA Database

9.1 Column-Based and Row-Based Storage in the SAP HANA Database

The SAP HANA database supports both row-based and column-based storage. However, it is optimized for column storage.

Tables that are organized in columns are read optimized and have better compression rates than tables organized in rows. Furthermore, some features of the SAP HANA database, such as partitioning, are available only for column tables. Column-based storage is typically suitable for big tables with bulk updates. However, update and insert performance is better on row tables. Row-based storage is typically suitable for small tables with frequent single updates.

The following table outlines the criteria that you can use to decide whether to store your data tables as column tables or row tables:

<table>
<thead>
<tr>
<th>Storage Type</th>
<th>When to Use</th>
</tr>
</thead>
</table>
| Column store | • Calculations are typically executed on individual or a small number of columns.  
• The table is searched based on the values of a few columns.  
• The table has a large number of columns.  
• The table has a large number of rows and columnar operations are required (aggregate, scan, and so on)  
• High compression rates can be achieved because the majority of the columns contain only a few distinct values (compared to the number of rows). |
| Row store    | • The application needs to process only one single record at one time (many selects and/or updates of single records).  
• The application typically needs to access the complete record.  
• The columns contain mainly distinct values so compression rate would be low.  
• Neither aggregations nor fast searching are required.  
• The table has a small number of rows (for example, configuration tables). |

Note:

• The SAP HANA database allows row tables to be joined with column tables. However, it is more efficient to join tables of the same storage type.  
• It is possible to change an existing table from one storage type to the other (ALTER TABLE ALTER TYPE).

Related Links

SAP HANA SQL Reference
9.1.1 Creating Tables

A table is a two dimensional data structure with cells organized in rows and columns. In order to load data into the SAP HANA database, you need to create tables. The tables can be row store or column store depending on the use case.

To create a table, you must be authorized to create objects in the selected schema.

1. In the Navigator view, open the catalog schema in which you want to create the new table.
2. In the context menu of the schema in which you want to create the table, choose New Table.
3. Enter the following information:
   - Table name
   - Table type (column store or row store)
4. Define the columns of your table as follows:
   a) Enter the name and properties of the first column.
   b) To add further columns, choose the button.
5. If necessary, add indexes.
   a) On the Indexes tab, choose the button.
   b) Specify the name and the index type (standard index or full-text index).
      A full-text index enables full-text search.
   c) In the lower part of the screen, define the index for the required column(s), together with any other necessary parameters.

   **Note:** You can create an index for a table any time either by right-clicking the table in the Navigator view and choosing New Index, or opening the table definition for editing.

Indexes are added to the table definition and in the schema’s Indexes folder.

6. To create the table, choose (Create Table).

The table appears in the Tables folder of the relevant schema.

**Note:** For more information creating and defining tables, loading data into tables, creating full-text indexes, and so on, see the SAP HANA Developer Guide and the SAP HANA SQL Reference.

Related Links

- SAP HANA Developer Guide
- SAP HANA SQL Reference

9.1.2 Creating Views

A view is a combination or selection of data from tables modeled to serve a particular purpose. Views appear like readable tables, in other words, database operations that read from tables can also be used to read data from views. For example, you can create a view that simply selects some columns from a table, or a view that selects some columns and some rows according to a filter pattern.

To create a view, you must be authorized to create objects in the selected schema and to select data from the tables to be included in the view.

1. In the Navigator view, open the catalog and navigate to the Views folder in the relevant schema.
2. In the context menu, choose New View. The editor for creating a new view opens.

3. Specify the view name.

4. Select the relevant tables by dragging them from the Navigator view into the editor area, or by choosing the + (Insert) button in the editor.

5. To create a join, proceed as follows:
   a) Drag a column from one table to the column of another table.
   b) Choose the join type in the Join Order area.
   If you define more than one join, you can define the order in which the joins are executed using drag and drop.

6. Drag and drop the columns to be contained in the result set into the Columns area. You can specify additional constraints or create synonyms for column names here.

7. To preview the data, choose Data Preview in the context menu of the editor.

8. To show the equivalent SQL statement, choose Export SQL in the context menu of the editor.

9. To create the view, choose the green circle (Execute) button. The view appears in the Views folder of the relevant schema.

Related Links

SAP HANA Developer Guide

9.2 Memory Management in the Column Store

The column store is the part of the SAP HANA database that manages data organized in columns in memory. Tables created as column tables are stored here. The column store is optimized for read operations but also provides good performance for write operations. This is achieved through 2 data structures: main storage and delta storage.

The main storage contains the main part of the data. Here, efficient data compression is applied to save memory and speed up searches and calculations. Write operations on compressed data in the main storage would however be costly. Therefore, write operations do not directly modify compressed data in the main storage. Instead, all changes are written to a separate data structure called the delta storage. The delta storage uses only basic compression and is optimized for write access. Read operations are performed on both structures, while write operations only affect the delta.

Figure 8: Main Storage and Delta Storage
The purpose of the delta merge operation is to move changes collected in the delta storage to the read-optimized main storage. After the delta merge operation, the content of the main storage is persisted to disk and its compression recalculated and optimized if necessary.

A further result of the delta merge operation is truncation of the delta log. The delta storage structure itself exists only in memory and is not persisted to disk. The column store creates its logical redo log entries for all operations executed on the delta storage. This log is called the delta log. In the event of a system restart, the delta log entries are replayed to rebuild the in-memory delta storages. After the changes in the delta storage have been merged into the main storage, the delta log file is truncated by removing those entries that were written before the merge operation.

**Note:** As only data in memory is relevant, the load status of tables is significant. A table can have one of the following load statuses:

- Unloaded, that is, none of the data in the table is loaded to main memory
- Partly loaded, that is, some of the data in the table is loaded to main memory, for example, a few columns recently used in a query
- Fully loaded, that is, all the data in the table is loaded into main memory

However, data that is in the delta storage can only be fully loaded or unloaded. Partial loading is not possible. Therefore, if a delta merge has not been performed and the table’s entire data is in the delta storage, the table is either fully loaded or unloaded.

### 9.2.1 Loading and Unloading Column Tables into and from Memory

Under normal circumstances, the SAP HANA database manages the loading and unloading of tables into and from memory independently – the aim being to keep all relevant data in memory. However, you can manually load and unload individual tables and table columns if necessary.

To load a table into memory, you must have the UPDATE SQL privilege for the table.

The SAP HANA database aims to keep all relevant data in memory. Standard row tables are loaded into memory when the database is started and remain there as long as it is running. They are not unloaded. Column tables, on the other hand, are loaded on demand, column by column when they are first accessed. This is sometimes called lazy loading. This means that columns that are never used are not loaded and memory waste is avoided.

**Note:** This is the default behavior of column tables. In the metadata of the table, it is possible to specify that individual columns or the entire table are loaded into memory when the database is started.
The database may also actively unload tables or individual columns from memory, for example, if a query or other processes in the database require more memory than is currently available. It does this based on a least recently used algorithm.

As the SAP HANA database automatically manages the loading and unloading of tables, you should normally not have to interfere with this process. However, you can manually load and unload individual tables and table columns if necessary. For example:

- To precisely measure the total or “worst case” amount of memory used by a particular table (load)
- To actively free up memory (unload)

**Note:** You can see detailed information about a table’s current memory usage and load status by viewing its definition in the table editor (*Runtime Information* tab).

<table>
<thead>
<tr>
<th>What</th>
<th>How</th>
</tr>
</thead>
</table>
| Loading and unloading tables using menu command in SAP HANA studio | 1. In the *Navigator* view, navigate to the table.  
2. In the context menu of the table, choose *Load* or *Unload* as required.  
3. Choose *OK*. |
| Loading and unloading tables using SQL    | Open the SQL console and execute the required statement:  
- LOAD <table_name>  
- UNLOAD <table_name> |
| Loading and unloading individual columns using SQL | Open the SQL console and execute the required statement:  
- LOAD <table_name> (<column_name>, ...)  
- UNLOAD <table_name> (<column_name>, ...) |

If you loaded a table, the complete data of the table, including the data in its delta storage, is loaded into main memory. Depending on the size of the table, this may take some time. The table’s load status is FULL.

If you unloaded a table, the complete data of the table, including the data in its delta storage, is unloaded from main memory. Subsequent access to this table will be slower as the data has to be reloaded into memory. The table’s load status is NO.

If you loaded or unloaded a column, the entire column is loaded or unloaded into or from main memory. Its load status is TRUE or FALSE. The table’s load status is PARTIALLY.

You can verify the load status of a table and its columns by opening the table in the table editor and referring to the *Runtime Information* tab.

**Related Links**

*Memory Usage of Column and Row Tables* [page 95]

A significant part of the used memory value in the SAP HANA database is the space used by data tables. Separate measurements are available for column-store tables and row-store tables.

*Memory Management in the Column Store* [page 104]
The column store is the part of the SAP HANA database that manages data organized in columns in memory. Tables created as column tables are stored here. The column store is optimized for read operations but also provides good performance for write operations. This is achieved through 2 data structures: main storage and delta storage.

Table Editor Views for Individual Table Analysis [page 121]
To analyze or monitor aspects of an individual table or view, you can open it in the table editor. There are different views available depending on what you want to do.

9.3 The Delta Merge Operation

Write operations are only performed on the delta storage. In order to transform the data into a format that is optimized in terms of memory consumption and read performance, it must be transferred to the main storage. This is accomplished by the delta merge operation.

The following figure shows the different steps in the merge process, which objects are involved, and how they are accessed:

Figure 9: The Delta Merge Process

1. Before the merge operation, all write operations go to Delta 1 storage and all read operations read from Main 1 and Delta 1 storages.
2. While the merge operation is running, the following happens:
   a. All write operations go to the second delta storage, Delta 2.
   b. Read operations read from the original main storage, Main 1, and from both delta storages, Delta 1 and Delta 2.
   c. Uncommitted changes in Delta1 are copied to Delta2.
   d. The content of Main 1 and the committed entries in Delta 1 are merged into the new main storage, Main 2.
3. After the merge operation has completed, the following happens:
   a. Main1 and Delta1 storages are deleted.
   b. The compression of the new main storage (Main 2) is reevaluated and optimized. If necessary, this operation reorders rows and adjust compression parameters. If compression has changed, columns are immediately reloaded into memory.
   c. The content of the complete main storage is persisted to disk.
d. The delta log is truncated.

Note: With this double buffer concept, the table only needs to be locked for a short time: at the beginning of the process when open transactions are moved to Delta2, and at the end of the process when the storages are “switched”.

Caution: The minimum memory requirement for the delta merge operation includes the current size of main storage + future size of main storage + current size of delta storage + some additional memory. It is important to understand that even if a column store table is unloaded or partly loaded, the whole table is loaded into memory to perform the delta merge.

The delta merge operation can therefore be expensive for the following main reasons:

- The complete main storages of all columns of the table are re-written in memory. This consumes some CPU resources and at least temporarily duplicates the memory needed for the main storages (while Main 1 and Main 2 exist in parallel).
- The complete main storages are persisted to disk, even if only a relatively small number of records were changed. This creates disk I/O load.

This potentially negative impact on performance can be mitigated by the following strategies:

- Executing memory-only merges
  A memory-only merge affects only the in-memory structures and does not persist any data.
- Splitting tables
  The performance of the delta merge depends on the size of the main storage. This size can be reduced by splitting the table into multiple partitions, each with its own main and delta storages. The delta merge operation is performed at partition level and only for partitions that actually require it. This means that less data needs to be merged and persisted. Note that there are disadvantages to partitioning tables that should also be considered. For more information, see the documentation on partitioning and distribution of tables.

Related Links

Monitoring Delta Merge Processing in Column Store Tables
Table Partitioning in the SAP HANA Database [page 134]

The partitioning feature of the SAP HANA database makes it possible to split column-store tables horizontally into disjunctive sub-tables or partitions. In this way, very large tables can be broken down into smaller, more manageable parts. Partitioning is typically used in distributed systems, but it may also be beneficial for single-host systems.

9.3.1 Merge Motivations

The request to merge the delta storage of a table into its main storage can be triggered in several ways. These are called merge motivations.

Auto Merge

The standard method for initiating a merge in SAP HANA is the auto merge. A system process called mergedog periodically checks the column store tables that are loaded locally and determines for each individual table (or single partition of a split table) whether or not a merge is necessary based on configurable criteria (for example, size of delta storage, available memory, time since last merge, and others).

Auto merge is active if the active parameter in the mergedog section of the indexserver.ini file is set to yes.
**Smart Merge**

If an application powered by SAP HANA requires more direct control over the merge process, SAP HANA supports a function that enables the application to request the system to check whether or not a delta merge makes sense now. This function is called smart merge. For example, if an application starts loading relatively large data volumes, a delta merge during the load may have a negative impact both on the load performance and on other system users. Therefore, the application can disable the auto merge for those tables being loaded and send a “hint” to the database to do a merge once the load has completed.

When the application issues a smart merge hint to the database to trigger a merge, the database evaluates the criteria that determine whether or not a merge is necessary. If the criteria are met, the merge is executed. If the criteria are not met, the database takes no further action and only a subsequent hint from the application will trigger another evaluation of the criteria.

Auto merge is active if the smart_merge_enabled parameter in the mergedog section of the `indexserver.ini` file is set to yes.

**Hard and Forced Merges**

You can trigger the delta merge operation for a table manually by executing the SQL statement `MERGE DELTA OF '<table_name>'`. This is called a hard merge and results in the database executing the delta merge for the table either immediately if sufficient system resources are available, or as soon as sufficient system resources become available. The hard merge is therefore subject to the merge token control mechanism.

If you want the merge to take place immediately regardless of system resource availability, you can pass an optional parameter. A forced merge may be useful in a situation where there is a heavy system load, but a small table needs to be merged or if a missed merge of a certain table is negatively impacting system performance. To execute a forced merge, execute the SQL statement `MERGE DELTA OF '<table_name>' WITH PARAMETERS ('FORCED_MERGE' = 'ON')`.

A further optional parameter allows you to tell the system to do the merge in memory only. This means that the final step of the delta merge, persisting main storage to disk, is omitted. This parameter can therefore be used to reduce the memory consumption caused by many entries in the delta storage by performing a merge but without causing any I/O load.

**Example:** During a mass data import, you trigger multiple memory-only merge operations by SQL. When the import is complete, you trigger one final “full” delta merge. In this way, the content of the table is only persisted once.

The main disadvantage of the memory-only merge is that the delta log is not truncated. In the event of a system restart, this may cause long delta log replay times.
To execute a memory-only merge, execute the SQL statement `MERGE DELTA OF '<table_name>' WITH
PARAMETERS ('MEMORY_MERGE' = 'ON')`

1. **Note:** Unlike system-triggered delta merge operations, all of the above manually-executed delta merge operations do not subsequently trigger an optimization of the compression of the table's new main storage. If the table was compressed before the delta merge operation, it remains compressed with the same compression strategy afterwards. If it was not compressed before the delta merge operation, it remains uncompressed afterwards. After a manual delta merge, you must therefore trigger compression optimization manually.

---

**Critical Merge**

The database can trigger a critical merge in order to keep the system stable. For example, in a situation where auto merge has been disabled and no smart merge hints are sent to the system, the size of the delta storage could grow too large for a successful delta merge to be possible. The system initiates a critical merge automatically when a certain threshold is passed. Critical merge is inactive by default.

**Related Links**

- **Performing a Manual Delta Merge Operation** [page 115]
  - You can trigger the delta merge operation for a column table manually in the SAP HANA studio, for example, if you need to free up memory.

- **Compressing Column Tables Manually** [page 119]
  - The SAP HANA database decides which columns in a column table to compress and which compression algorithm to apply for each column. It does this as part of the delta merge operation. It is normally not necessary or recommended that you interfere with this process. However, you can trigger compression manually.

---

**SAP HANA Database – System Tables and Monitoring Views**

### 9.3.2 The Merge Monitor

The delta merge operation for column tables is a potentially expensive operation and must be managed according to available resources and priority. This is the responsibility of the merge monitor.

The system uses cost functions to decide which table to merge, when, and in which order. There are also cost functions that control how many tables are merged at the same time and how many threads are used to merge a single table.

The merge monitor is responsible for controlling all merge requests for all column tables on a single host. In a distributed system, every index server has its own merge monitor.

All merge requests must acquire a merge token from the merge monitor. A merge token represents an allocation of system resources and "entitles" the merge to actually start. The merge monitor blocks merge requests if there are not enough system resources available or if the same table is already being merged by another thread. This avoids long waits and delays for other threads for inserting or just reading data.

Depending on current system resource consumption, merge motivation, and the evaluation of the various cost functions, the merge monitor lets single requesting merge threads pass and releases waiting threads.

1. **Note:** There is no option or need to disable, stop, or even kill the merge monitor. The merge monitor is not a thread.
9.3.3 Cost Functions

The SAP HANA database decides whether or not to execute a requested delta merge and the order in which to execute multiple requests based on configurable merge criteria or cost functions.

Cost functions can be configured depending on the merge motivation, that is whether the merge is being requested by the automatic system process mergedog (auto merge), by a hint from the application (smart merge), by SQL statement (hard merge), and so on.

Cost functions are evaluated in runtime and configured in the mergedog section of the `indexserver.ini` file. The following cost functions are available:

- **auto_merge_decision_function** and **smart_merge_decision_func**
  These cost functions determine whether or not a requested delta merge is executed.

- **auto_merge_priority_func** and **smart_merge_priority_func**
  These cost functions determine the priority that is assigned to the delta merge request.

- **hard_merge_priority_func**
  This cost function determines the priority of hard merges.

- **load_balancing_func**
  This cost function determines the allocation of system resources to merge processing.

**Note:** The decision cost function is evaluated only once for each merge request. In the case of a merge request triggered by a smart merge hint, if the cost function returns a result of false (that is, the system decides that a delta merge is not required), the request is logged but no further evaluation takes place. Only a new hint can potentially initiate a new delta merge.

The following parameters are available for configuring the cost functions. You can use them to build cost functions for all delta merge configurations.

**Caution:** It is not recommended that you change the default settings for delta merge unless instructed to do so by SAP Support.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Meaning</th>
</tr>
</thead>
</table>
| DMS       | Delta memory size [MB]  
This refers to the size of the table's delta storage. |
| TMD       | Table merge delay [sec]  
This refers to how long since the last delta merge |
| MRC       | Main row count [million]  
This refers to the current number of rows in the main storage of the table. |
| DMR       | Deleted main rows  
This refers to the number of deleted records not in delta storage, but marked as deleted in main storage. Merging makes sense if there are many deleted rows. |
<p>| DLS       | Delta log size [MB] |
| DCC       | Delta cell count [million] |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>This refers to the current number of cells in the delta storage of the table. For example, if the delta storage contains 3 records, each with 4 columns, then the delta cell count is 12.</td>
</tr>
<tr>
<td>DRC</td>
<td>Delta row count [million]</td>
</tr>
<tr>
<td></td>
<td>This refers to the current number of rows in the delta storage of the table.</td>
</tr>
<tr>
<td>QDW</td>
<td>Queuing delay wait [sec]</td>
</tr>
<tr>
<td></td>
<td>This refers to the time that a merge thread has been waiting for the merge monitor to allocate it merge tokens. This parameter can be useful if you want to implement a first come first served scheduling strategy.</td>
</tr>
<tr>
<td>NAME</td>
<td>Table name [string]</td>
</tr>
<tr>
<td>SCHEMA</td>
<td>Schema name [string]</td>
</tr>
<tr>
<td>CLA</td>
<td>CPU load average [percentage]</td>
</tr>
<tr>
<td>LCC</td>
<td>Logical CPU count</td>
</tr>
<tr>
<td>THM</td>
<td>Total heap memory [MB]</td>
</tr>
<tr>
<td>AHM</td>
<td>Available heap memory, including memory that could be freed [MB]</td>
</tr>
</tbody>
</table>

### Cost Functions Examples

<table>
<thead>
<tr>
<th>Cost Function Configuration</th>
<th>Meaning</th>
</tr>
</thead>
</table>
| auto_merge_decision_function = DMS>1000 or TMD>3601 or DCC>800 or DMR>0.2*MRC or DLS>5000 | An automatic delta merge of a table is executed if:  
  - The size of its delta storage exceeds 1000 MB, or  
  - It has not been merged in over 60 minutes, or  
  - Its delta cell count exceeds 800 million, or  
  - More than 20% of the records in its main storage were deleted, or  
  - The size of its delta log is greater than 5000 MB |
| auto_merge_decision_func = DMS > 1000 or DMS > 42 and weekday(now())=6 and secondtime(now())>secondtime('01:00') and secondtime(now())<secondtime('02:00') | An automatic delta merge of a table is executed if:  
  - The size of its delta storage exceeds 1000 MB, unless  
  - It is Saturday between 1.00 and 2.00, in which case it will be merged if delta storage exceeds 42MB  
  Note the week starts with Monday as day 0. |
| smart_merge_decision_func = DMS>1000 or DCC>800 or DLS>5000 | A delta merge request of a table triggered by a smart merge hint is executed if:  
  - The delta storage size exceed 1000 MB, or |
### Cost Function Configuration

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Meaning</th>
</tr>
</thead>
</table>
| smart_merge_priority_func = DMS/1000 | - The delta cell count in the delta storage is greater than 800 million, or  
  - The size of the delta log is greater than 5000 MB  
  The system prioritizes smart merge requests based on the size of the delta storage, that is, tables with the bigger deltas are merged first. |
| hard_merge_priority_func = QDW    | Delta merges triggered by hard merge are prioritized only by queuing delay wait, in other words, on a first in first out basis. |
| hard_merge_priority_function = 1/(7+MMS) | Delta merges triggered by hard merge are prioritized by table size. Smaller tables are merged first, the idea being to free some memory first before bigger tables start merging. |

### Related Links

*Changing System Properties* [page 32]

The properties of an SAP HANA system are defined in the parameters of its configuration files. Configuration files are separated into sections; sections bundle parameters of the same category.

### 9.3.4 Merge Tokens

The delta merge operation can create a heavy load on the system. Therefore, controls need to be applied to ensure that merge operations do not consume all system resources. The control mechanism is based on the allocation of merge tokens to each merge operation.

With the exception of the forced merge, a merge operation cannot start unless it has been allocated tokens. If all merge tokens are taken, merge requests have to wait either until the system releases new merge tokens because more resources are available, or until merge tokens have been released by completed merge requests.

The number of merge tokens available for allocation is adjusted based on current system resource availability. This number is recalculated periodically by the system based on a cost function configured in the `load_balancing_func` parameter in the `mergedog` section of the `indexserver.ini` file. The default configuration is `load_balancing_func = 1 + LCC * (100-CLA)/100`. If a hard maximum is required for the amount of tokens available, you can configure a constant value configured or a constant parameter (for example, LCC). Each merge token represents a single CPU.

For every merge request, the number of tokens required to perform the merge is calculated by the system. If the system is not able to determine a value, a default value is returned. This default value can be configured in the `token_per_table` parameter in the `mergedog` section of the `indexserver.ini` file. However, it is not recommended that you change this value.

**Note:** It is not possible to check the number of merge tokens available for allocation at any given time, but it is logged in the `indexserver` trace file if you activate the `indexserver` component `mergemonitor` with trace level `INFO`.

### Related Links

*Working with Diagnosis Files* [page 220]
In the event of problems with the SAP HANA database, you can display the relevant diagnosis file and analyze for errors. You can also filter, merge, delete, and download diagnosis files.

### 9.3.5 Monitoring Delta Merge History

Information about all delta merge operations since the last system start are logged in the monitoring view `M_DELTA_MERGE_STATISTICS`. In addition to completed merge operations, information is available on merge hints received by applications and post-merge compression optimization.

You can access a predefined view of these merge statistics in the Administration editor on the `System Information` tab.

The following columns contain potentially useful information:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
</table>
| TYPE     | Here you can see the type of merge history entry. The following values are possible:  
- MERGE for an actual delta merge operation  
- HINT for a merge hint sent to SAP HANA by an application  
- SPARSE for the post-merge optimization of main storage compression |
| MOTIVATION | This column identifies the underlying merge motivation: AUTO, SMART, HARD, or FORCE |
| SUCCESS  | This column depends on the entry in the TYPE column.  
- For MERGE or SPARSE entries, it indicates whether or not the merge or compression optimization operation was successful.  
- For HINT entries, it indicates whether or not the hint from the application to merge was accepted.  
  
  If the hint was accepted (SUCCESS=TRUE), then there is an associated entry of type MERGE. If the hint was rejected (SUCCESS=FALSE), then no merge is triggered, so there is no associated MERGE entry. |
| LAST_ERROR | This column provides information about errors.  
The following error codes are possible:  
- Error 2480: The table in question is already being merged.  
- Error 2481: There are already other smart merge requests for this table in the queue.  
- Error 2482: The delta storage is empty or the evaluation of the smart merge cost function indicated that a merge is not necessary.  
- Error 2483: Smart merge is not active (parameter smart_merge_enabled=no).  
- Error 2484: Memory required to optimize table exceeds heap limit (for failed compression optimization operations (TYPE=SPARSE, SUCCESS=FALSE)). |
| PASSPORT | For entries with the merge motivation SMART, this column identifies the application that sent the hint to merge (for example, SAP NetWeaver BW powered by SAP HANA) |

**Note:** Even if the hint was accepted (SUCCESS=TRUE), this does not necessarily mean that the subsequent merge was successful. You must check the SUCCESS column of the merge entry.

**Note:** If the index server is restarted, the delta merge history will initially be empty. The statistics server also collects delta merge statistics in the table `HOST_DELTA_MERGE_STATISTICS` independent of
Example: The following is an example of how to use the merge history to find a merge you were expecting to happen based on the settings for triggering smart merge hints in your application.

1. Look for merges triggered by smart merge in the merge history by executing the following SQL statement:

   ```sql
   SELECT * FROM M_DELTA_MERGE_STATISTICS WHERE table_name = '<your_table>'
   AND motivation = 'SMART'
   ```

2. If no results are returned, check to see if the application actually sent any hints by executing the following statement:

   ```sql
   SELECT * FROM M_DELTA_MERGE_STATISTICS WHERE type = 'HINT' AND table_name = '<your_table>'
   ```

   If the application did not send a hint, then the system will not initiate a delta merge. However, if the application did send a hint, the system only executes the merge if the criteria for smart merge are fulfilled. The information is available in the SUCCESS column. The system decides whether or not to accept the hint and execute the merge by evaluating the smart merge decision cost function.

3. If you still have not found the smart merge, check the long-term history by executing the following statement:

   ```sql
   SELECT * FROM _SYS_STATISTICS.HOST_DELTA_MERGE_STATISTICS WHERE table_name = 'your_table'
   ```

Tracing

You can activate the logging of merge-related information in the database trace for the indexserver component. The relevant trace components are mergemonitor and mergedog. We recommend the trace level INFO.

Related Links

- SAP HANA System Tables and Monitoring Views Reference
- Working with Diagnosis Files [page 220]
  In the event of problems with the SAP HANA database, you can display the relevant diagnosis file and analyze for errors. You can also filter, merge, delete, and download diagnosis files.
- Configuring Traces [page 221]
  You can activate and configure several traces on the Trace Configuration tab of the Administration editor. Trace data is saved to trace files, which you can view on the Diagnosis Files tab.

9.3.6 Performing a Manual Delta Merge Operation

You can trigger the delta merge operation for a column table manually in the SAP HANA studio, for example, if you need to free up memory.

To perform a delta merge operation for a table manually, you must have the UPDATE SQL privilege for the table.

It may be necessary or useful to trigger a merge operation manually in some situations, for example:

- An alert has been issued because a table is exceeding the threshold for the maximum size of delta storage.
- You need to free up memory. Executing a delta merge operation on tables with large delta storages is one strategy for freeing up memory. The delta storage does not compress data well and it may hold old versions of records that are no longer required for consistent reads. For example, you can use the following SQL
statement to retrieve the top 100 largest delta storages in memory:

```sql
SELECT TOP 100 * FROM M_CS_TABLES ORDER BY MEMORY_SIZE_IN_DELTA DESC.
```

You can trigger the delta merge operation for a column table manually in the SAP HANA studio by menu command or SQL statement. A manually-executed delta merge operation corresponds to a hard merge. However, if you use SQL, you can also pass additional parameters that trigger forced merges and memory-only merges.

1. Execute the required merge in one of the following ways:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Menu command** | 1. In the Navigator view, navigate to the table.  
2. In the context menu of the table, choose Merge.  
3. Choose OK. |
| **SQL**   | Open the SQL console and execute the required statement:  
○ MERGE DELTA OF '<table_name>' (hard merge)  
○ MERGE DELTA OF '<table_name>' WITH PARAMETERS ('FORCED_MERGE' = 'ON') (forced merge)  
○ MERGE DELTA OF '<table_name>' WITH PARAMETERS ('MEMORY_MERGE' = 'ON') (memory-only merge) |

2. Optional: Confirm the delta merge operation in one of the following ways:

- Open the table definition in the table editor and on the Runtime Information tab, check the relevant values

**Note:** Even though the delta merge operation moves data from the delta storage to the main storage, the size of the delta storage will not be zero. This could be because while the delta merge operation was taking place, records written by open transactions were moved to the new delta storage. Furthermore, even if the data containers of the delta storage are empty, they still need some space in memory.

- Check the merge history by opening the Merge Statistics table on the System Information tab

The SUCCESS column indicates whether or not the merge operation was executed.

**Note:** Unlike system-triggered delta merge operations, manually-executed delta merge operations do not subsequently trigger an optimization of the compression of the table’s new main storage. If the table was compressed before the delta merge operation, it remains compressed with the same compression strategy afterward. If it was not compressed before the delta merge operation, it remains uncompressed afterward. After a manual delta merge, you must therefore trigger compression optimization manually.

### 9.3.7 Viewing Progress of Delta Merge Operations

You can see the progress of delta merge operations currently running in the Administration editor on the Performance Job Progress tab.

Here, progress is indicated as the number of steps already completed (CURRENT_PROGRESS) of the total number of steps (MAX_PROGRESS) in the operation. For the delta merge operation, the number of steps corresponds to the number of columns in the table.

**Related Links**
The Delta Merge Operation [page 107]
Write operations are only performed on the delta storage. In order to transform the data into a format that is optimized in terms of memory consumption and read performance, it must be transferred to the main storage. This is accomplished by the delta merge operation.

Monitoring System Performance [page 84]
In addition to the general information about the overall system performance that you can see in the System Monitor and on the Overview tab of the Administration editor, you can monitor more detailed aspects of system performance on the Performance tab.

9.4 Data Compression in the Column Store

The column store allows for the efficient compression of data. This makes it less costly for the SAP HANA database to keep data in main memory. It also speeds up searches and calculations.

Data in column tables can have a two-fold compression:

- Dictionary compression
  This default method of compression is applied to all columns. It involves the mapping of distinct column values to consecutive numbers, so that instead of the actual value being stored, the typically much smaller consecutive number is stored.

- Advanced compression
  Each column can be further compressed using different compression methods, namely prefix encoding, run length encoding (RLE), cluster encoding, sparse encoding, and indirect encoding. The SAP HANA database uses compression algorithms to determine which type of compression is most appropriate for a column.

Note: Only the main storage of column tables are compressed. As the delta storage is optimized for write operations, it is not compressed.

Compression is automatically calculated and optimized as part of the delta merge operation. If you create an empty column table, no compression is applied initially as the database cannot know which method is most appropriate. As you start to insert data into the table and the delta merge operation starts being executed at regular intervals, data compression is automatically (re)evaluated and optimized.

Automatic compression optimization is ensured by the parameter active in the optimize_compression section of the indexserver.ini configuration file. This parameter must have the value yes.

Compression Factor
The compression factor refers to the ratio of the uncompressed data size to the compressed data size in SAP HANA.

The uncompressed data volume is a database-independent value that is defined as follows: the nominal record size multiplied by the number of records in the table. The nominal record size is the sum of the sizes of the data types of all columns.

The compressed data volume in SAP HANA is the total size that the table occupies in the main memory of SAP HANA.
Example: You can retrieve this information for a fully-loaded column table from the monitoring view M_CS_TABLES by executing the statement:

```sql
select SCHEMA_NAME, TABLE_NAME, MEMORY_SIZE_IN_TOTAL from PUBLIC.M_CS_TABLES where SCHEMA_NAME='<schema>' and TABLE_NAME='<table>'
```

The compression factor achieved by the database depends on your SAP HANA implementation and the data involved.

Related Links
- SAP Note 1514966 (SAP HANA 1.0: Sizing SAP In-Memory Database)
- SAP Note 1637145 (SAP BW on HANA: Sizing SAP In-Memory Database)

9.4.1 Checking Compression of Column Tables

For column-store tables, you can check the type of compression applied to table columns, as well as the compression ratio.

To check the compression status of a table accurately, ensure that it is first fully loaded into main memory.

<table>
<thead>
<tr>
<th>What</th>
<th>How</th>
</tr>
</thead>
</table>
| Checking the type of compression applied to table columns | 1. In the Administration editor, open the SQL console.  
2. Execute the following SQL statement:  
```sql
SELECT SCHEMA_NAME, TABLE_NAME, COLUMN_NAME, COMPRESSION_TYPE, LOADED from PUBLIC.M_CS_COLUMNS where SCHEMA_NAME='<your_schema>' and TABLE_NAME='<your_table>'
```

The columns of the selected table are listed with the type of compression applied. The following values are possible:
- DEFAULT
- SPARSE
- PREFIXED
- CLUSTERED
- INDIRECT
- RLE

Note: Even if the column is not loaded into memory, the compression type is indicated as DEFAULT. This is because there will always be some level of dictionary compression. However, unless the column is loaded, the database cannot determine the type of compression actually applied. The LOADED column indicates whether or not the column is loaded into memory.
Checking the compression ratio of table columns, that is, the ratio of the column’s uncompressed data size to its compressed data size in memory.

1. In the Administration editor, open the table definition in the table editor.
2. Choose the Runtime Information tab.
3. In the Details for Table area, choose the Columns tab.

The compression ratio is listed in the Main Size Compression Ratio [%] column.

### Related Links

*Loading and Unloading Column Tables into and from Memory* [page 105]

Under normal circumstances, the SAP HANA database manages the loading and unloading of tables into and from memory independently – the aim being to keep all relevant data in memory. However, you can manually load and unload individual tables and table columns if necessary.

### 9.4.2 Compressing Column Tables Manually

The SAP HANA database decides which columns in a column table to compress and which compression algorithm to apply for each column. It does this as part of the delta merge operation. It is normally not necessary or recommended that you interfere with this process. However, you can trigger compression manually.

To compress a table manually, you must have the UPDATE privilege for the table.

We do not recommend that you interfere with the way in which the SAP HANA database applies compression. However, if a table is not compressed and you think it should be, you can request the database to reevaluate the situation.

Before you do this, consider the reasons why the table may not be compressed, for example:

- The table is very small.
- The table’s delta storage has never been merged with its main storage.
- The table was created and filled using an old version of the SAP HANA database that did not compress data automatically. No further data loads, and consequently no delta merge operations, have taken place.

1. In the Administration editor, open the SQL console.
2. Request the database to reevaluate compression by executing the SQL statement `UPDATE '<your_table>' WITH PARAMETERS ('OPTIMIZE_COMPRESSION'='YES')`. The database checks all of the table’s columns and determines whether or not they need to be compressed, or whether or not existing compression can be optimized. If this is the case, it compresses the data using the most appropriate compression algorithm. However, note the following:
   - The database will only reevaluate compression if the contents of the table have changed significantly since the last time compression was evaluated.
   - Even if the database does reevaluate the situation, it may determine that compression is not necessary or cannot be optimized and so changes nothing.
3. Check the compression status of the table.
4. Optional: If compression has not changed, force the database to reevaluate compression by executing the following SQL statement:

```
UPDATE '<your_table>' WITH PARAMETERS
('OPTIMIZE_COMPRESSION'='FORCED')
```

The database checks all of the table’s columns and determines whether or not they need to be compressed, or whether or not existing compression can be optimized. If this is the case, it compresses the data using the most appropriate compression algorithm. Note that the database may still determine that compression is not necessary or cannot be optimized and so changes nothing.

5. Check the compression status of the table.

Related Links

*The Delta Merge Operation* [page 107]

Write operations are only performed on the delta storage. In order to transform the data into a format that is optimized in terms of memory consumption and read performance, it must be transferred to the main storage. This is accomplished by the *delta merge operation*.

### 9.5 Opening Tables in the Table Editor

Some monitoring and problem analysis may require you to examine individual tables, for example, the many system tables and monitoring views provided by the SAP HANA database. You can open tables and views directly in the table editor. Different viewing options are available depending on what you want to do.

You can open tables and views in the table editor in the following ways:

<table>
<thead>
<tr>
<th>Option</th>
<th>Steps</th>
</tr>
</thead>
</table>
| **Navigate** | 1. In the *Navigator* view, navigate to the table or view you want to open.  
2. From the context menu, choose which view you want to open:  
   ○ Table definition  
   ○ Table content  
   ○ Data preview  
   ![Note](https://via.placeholder.com/150) **Note:** By default, double-clicking the table in the *Navigator* view opens its definition. You can configure this setting in the Administration Console’s preferences. |
| **Search** | 1. From the Navigator view toolbar, choose the *(Find Table)* button.  
2. Enter a search string (at least two characters).  
3. Select the required table.  
   Matching tables are displayed immediately.  
4. Choose whether you want to display the table content and/or the table definition. |

The table is displayed in the table editor.
9.5.1 Table Editor Views for Individual Table Analysis

To analyze or monitor aspects of an individual table or view, you can open it in the table editor. There are different views available depending on what you want to do.

Table Definition

The table definition view provides you with information about the table’s structure and properties (for example, schema, type, column properties, and indexes).

Detailed information relating to the table’s memory usage and size is available on the Runtime Information sub-tab. This information can be useful in the following cases, for example:

- You want to examine the memory usage of an individual table in detail as part of performance analysis or optimization.
- You want to review the partitioning of a table.

Due to the different memory management concepts for row store and column store tables, the information displayed varies according to table type.

For column-store tables, you can review the following information:

- Overall memory usage information for the table, including total size of the table in memory, size of main and delta storages in memory, number of records, and size on disk

  **Note:** It is not possible to accurately determine the memory consumption of a table from its size on disk. This is because not all data structures that represent a table are stored on disk, they are only created when the table is loaded into memory.

- Detailed memory usage information at the level of partition and individual column

  **Note:** If the table is not partitioned, the information for the single item on the Parts tab is for the table.

The following information may be useful:

- **Total size**
  
  The cumulative in-memory size of all columns and internal structures in the partition, or of the individual column

- **Main size**
  
  The cumulative in-memory size of all columns in the partition in main storage, or of the individual column in main storage

- **Delta size**
  
  The cumulative in-memory size of all columns in the partition in delta storage, or of the individual column in delta storage

- **Estimated maximum size**
The estimated maximum size of the table when loaded into memory, including main and delta storages

- Time of last delta merge operation
- Load status

Partitions can be fully, partially, or not loaded. Individual columns can be either loaded or not.

- Main storage compression ratio (%)

The current compression ratio of the column in main storage

Note: If you want to analyze the compression ratio of a table, it must be fully loaded into memory.

For row-store tables, you can review the following information:

- Overall memory usage information for the table, including total size of the table in memory, number of records, and size on disk
- Memory usage of fixed and variable parts of the table

Row tables are permanently stored in memory using a linked list of pages. The values displayed here indicate the occupancy level of available pages.

Note: The memory usage information of column and row store information displayed on the Runtime Information tab is retrieved from the following monitoring views:

- M_TABLES
- M_RS_TABLES
- M_CS_TABLES
- M_CS_COLUMNS
- M_CS_PARTITIONS

Table Content

The table content view shows the actual records in the table. For example, you want to view the content of a system table or monitoring view to help you understand what is happening in the database.

Note that by default, only the first 1000 rows are displayed. You can change this setting in the Administration Console’s preferences.

To view the full content of a table cell, for example a large object (LOB) value, in the context menu of the cell choose Export Cell to... Zoom...

Data Preview

The data preview view allows you to analyze the content of the table in different ways. Similarly to the table content view, this is particularly useful for analyzing system tables.

Example: You want to check the global memory consumption of the database over the last 30 days.

1. Open the data preview of the table GLOBAL_MEMORY_STATISTICS (_SYS_STATISTICS schema).
2. Choose the Data Analysis tab.
3. Move the column SERVER_TIMESTAMP from the Available Objects area to the Labels Axis area.
4. Move the column TOTAL_MEMORY_USED_BY_INSTANCE from the Available Objects area to the Values Axis area.
5. Choose your preferred graphical output.

Related Links
Memory Usage of Column and Row Tables [page 95]
A significant part of the used memory value in the SAP HANA database is the space used by data tables. Separate measurements are available for column-store tables and row-store tables.

9.6 Exporting Tables and Other Catalog Objects

You can export all catalog objects to a file system and then import them back into another database. For example, you want to move data from a test system to a productive system, clone your system, or provide the data to SAP Support so they can replicate a scenario.

To be able to export catalog objects, you must have the following privileges:

- System privilege EXPORT
- SQL privilege SELECT for the catalog objects in question

1. Select the objects you want to export in one of the following ways:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Find objects for export</td>
<td>1. From the main menu, choose File &gt; Export. The Export dialog box appears.</td>
</tr>
<tr>
<td></td>
<td>2. Search for the objects you want to export and add them to list of objects to be exported on the right.</td>
</tr>
<tr>
<td>Select object for export</td>
<td>1. In the Navigator view, select the objects that you want to export.</td>
</tr>
<tr>
<td></td>
<td>2. From the context menu, choose Export.</td>
</tr>
<tr>
<td></td>
<td>3. Optional: Search for additional objects to be exported and add them to list of objects to be exported on the right.</td>
</tr>
</tbody>
</table>

2. Choose Next.

3. Specify the scope of the export by choosing the relevant options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>The file format used for export can be either CSV or binary. Exports in binary format are faster and more compact. However, CSV format is better if you need to use the data in a non-HANA system. It also has the advantage of being human readable. CSV is selected by default.</td>
</tr>
</tbody>
</table>

Note: Column-store tables, procedures, and sequences can be exported in either binary or CSV format. However, row-store tables can be exported only in CSV format. If you
4. Specify the location to which the file is to be exported:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog and Data</td>
<td>The object definition and its data are exported. This option is selected by default.</td>
</tr>
<tr>
<td>Catalog Only</td>
<td>Only the object definition is exported. For example, you may want to copy only the table definition in order to create a new table with the same structure.</td>
</tr>
<tr>
<td>Including Dependencies</td>
<td>Dependent objects of selected objects are also exported, that is triggers and indexes. This option is selected by default.</td>
</tr>
</tbody>
</table>

4. Specify the location to which the file is to be exported:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Export Catalog Objects on Server</td>
<td>The selected catalog objects are saved to a directory on the database server file system. The default directory is /usr/sap/&lt;SID&gt;/&lt;instance&gt;/work.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If you want to specify a different directory in the server's file system, it must already exist and the database must be authorized to access it.</td>
</tr>
<tr>
<td>Export Catalog Objects to Current Client</td>
<td>The selected catalog objects are saved in the specified directory on the client file system.</td>
</tr>
</tbody>
</table>

5. Enter the number of parallel threads to be used for the export.

The more threads you use, the faster the export will be. This does however impact the performance of the database as more threads use more resources.

The following guidelines apply:

- For a view or procedure, use 2 or more threads, up to the number of dependent objects.
- For a whole schema, consider using more than 10 threads, up to the number of CPU cores in the system.
- For a whole SAP NetWeaver BW or SAP ERP system with tens of thousands of tables, using a large number of threads is reasonable (up to 256).


The catalog objects are exported to the specified location. Depending on the number of objects being exported and the scope of the export, this may take some time.

You can monitor the progress of a running export in the monitoring view M_EXPORT_BINARY_STATUS.

Information messages and errors are recorded in the Error Log view.

9.7 Importing Tables and Other Catalog Objects

You can import previously exported catalog objects into another database. For example, to move data from a test system to a productive system, clone your system, or provide the data to SAP Support so they can replicate a scenario.

To be able to import catalog objects, you must have the following privileges:
- System privilege IMPORT
- SQL privilege INSERT for the catalog objects in question

You can import previously exported catalog objects into another database. This may be necessary, for example, if you want to move data from a test system to a productive system, if you want to clone your system, or if you want to provide the data to SAP Support so they can replicate a certain scenario.

1. Select the objects you want to import in one of the following ways:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Find objects for import | 1. From the main menu, choose **File > Import**. The **Import** dialog box appears.  
2. Specify the location of previously exported objects:  
   ○ For objects exported on the database server file system, choose **Import Catalog Objects on Server** and enter the directory.  
   The default export directory is entered by default.  
   ○ For objects exported to a location on the client file system, choose **Import Catalog Objects from Current Client** and browse to the directory.  
3. Choose **Next**.  
4. Search for the objects that you want to import and add them to list of objects to be imported on the right. You can filter the search results by format type (CSV or binary). |

2. Choose **Next**.

3. Specify the scope of the import by choosing the relevant options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog and Data</td>
<td>The object definition and its data are imported.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This option is selected by default. However, the export must have included both definition and data.</td>
</tr>
<tr>
<td>Catalog Only</td>
<td>Only the object definition is imported. For example, you may want to copy only the table definition in order to create a new table with the same structure.</td>
</tr>
</tbody>
</table>

4. Enter the number of parallel threads to be used for the import.
The more threads you use, the faster the import will be. This does however impact the speed of the database as more threads uses more resources.

The following guidelines apply:

- For a view or procedure, use 2 or more threads, up to the number of dependent objects.
- For a whole schema, consider using more than 10 threads, up to the number of CPU cores in the system.
- For a whole SAP NetWeaver BW or SAP ERP system with tens of thousands of tables, using a large number of threads is reasonable (up to 256).

5. Choose Finish.

The catalog objects are imported from the specified location. Depending on the number of objects being imported and the scope of the import, this may take some time.

You can monitor the progress of a running import in the monitoring view M_IMPORT_BINARY_STATUS.

Information messages and errors are recorded in the Error Log view (Window ➤ Show View ➤ Error Log).
10 Scaling SAP HANA

10.1 About Scalability

There are two general approaches you can take to scale your SAP HANA system.

First, you can **scale up**. This means increasing the size of one physical machine by increasing the amount of RAM available for processing.

You can also **scale out**. This means combining multiple independent computers into one system. The main reason for distributing a system across multiple hosts (that is, scaling out) is to overcome the hardware limitations of a single physical server. This means that an SAP HANA system can distribute the load between multiple servers. In a distributed system, each index server is usually assigned to its own host to achieve maximum performance. It is possible to assign different tables to different hosts (partitioning the database), as well as to split a single table between hosts (partitioning of tables).

The following sections describe the various aspects of scalability.

### Scaling the Data

One technique you can use to deal with planned data growth is to purchase more physical RAM than is initially required, to set the allocation limit according to your needs, and then to increase it over time to adapt to your data. Once you have reached the physical limits of a single server, you can scale out over multiple machines to create a distributed SAP HANA system. You can do this by distributing different schemas and tables to different servers (complete data and user separation). However, this is not always possible, for example, when a single fact table is larger than the server’s RAM size.

The most important strategy for scaling your data is data partitioning. Partitioning supports the creation of very large tables (billions of rows) by breaking them into smaller chunks that can be placed on different machines. Partitioning is transparent for most SQL queries and other data manipulations.

### Scaling Performance

SAP HANA’s performance is derived from its efficient, parallelized approach. The more computation cores your SAP HANA server has, the better overall system performance.

Scaling performance requires a more detailed understanding of your workload and performance expectations. Using simulations and estimations of your typical query workloads, you can determine the expected load that a typical SAP HANA installation may comfortably manage. At the workload level, a rough prediction of scalability can be established by measuring the average CPU utilization while the workload is running. For example, an average CPU utilization of 45% may indicate that the system can be loaded 2X before showing a significant reduction in individual query response time.
Scaling the Application

Partitioning can be used to scale the application as it supports an increasing number of concurrent sessions and complex analytical queries by spreading the calculations across multiple hosts. Particular care must be taken in distributing the data so that the majority of queries match partitioning pruning rules. This accomplishes two goals: directing different users to different hosts (load balancing) and avoiding the network overhead related to frequent data joins across hosts.

Scaling Hardware

SAP HANA is offered in two ways – in the form of an appliance, delivered in a number of different configurations and "sizes" by certified hardware partners, or as part of a cloud-based service. This creates different system design options with respect to scale-up and scale-out variations. To maximize performance and throughput, SAP recommends that you scale up as far as possible (acquire the configuration with the highest processor and memory specification for the application workload), before scaling out (for deployments with even greater data volume requirements).

Note: The SAP HANA hardware partners have different building blocks for their scale-out implementations. Therefore, you should always consult with your hardware partner when planning your scale-out strategy.

10.2 Network Configuration

As part of setting up a distributed system you need to configure network parameters. The configuration of the network parameters should happen before you add additional hosts. This is done because one server needs to be available so that you can connect to the SAP HANA studio. In the Configuration tab of the SAP HANA studio you edit the section internal_hostname_resolution in the global.ini file. This is where the mapping of hostnames to IP addresses takes place.

Figure 10: General Network Layout
The figure shows a sample cluster with external addresses (10.68.22.*) and internal (192.168.2.*) addresses. To redirect the internal communication over the local network backbone you could map the internal addresses to the hostnames of SAP HANA servers as shown in this example:

```
[communication]
listeninterface = .internal
#listeninterface = .global
#listeninterface = .local
#listeninterface = 192.168.2.0/24

[internal_hostname_resolution]
192.168.2.101 = hana01
192.168.2.102 = hana02
192.168.2.103 = hana03
```

For increased security you can limit the binding of the processes in the communication section of the `global.ini` file. The option `listeninterface` can be set in two ways:

1. You can set it to one of the predefined keywords:
   - .global
   - .internal
   - .local
   
   **Note:**
   You must include the dot at the beginning of the keyword.

2. You can set it to a subnet in CIDR notation

The .global keyword (default) lets the process bind to all interfaces, or the local interfaces (the 127.*.*.*) interfaces. This configuration is only an option for single host installations as the server is not reachable from the outside. These two options do not require a valid `internal_hostname_resolution` section.

With the .internal setup an `internal_hostname_resolution` section is required. This configuration scans `internal_hostname_resolution` for the local address of the host. The process is only bound to this address (and to all localhost interfaces). So you should add all hosts and their respective addresses to the `global.ini` right after
the installation of the first server. Then the SAP HANA instance on the first server needs to be restarted for the changes to take effect. After that the remaining hosts may be added.

With this configuration the whole landscape uses the internal network right after installation. To reduce the possibility of errors it is also possible to install the whole landscape first without SAP HANA network configuration. This lets you run tests first before you establish the network. Then the configuration options remain the same and the whole SAP HANA landscape needs to be restarted for your changes to take effect.

**Related Links**

*SAP HANA Security Guide*

### 10.3 Perform a Distributed SAP HANA Setup

You have the possibility to add (or remove) additional hosts to your SAP HANA system with the On-Site Configuration tool. To do this, follow the procedure below:

1. Connect to the system with an X server client to enable GUI system access.
2. Open a root shell and go to the directory where you mounted the SAP HANA DVD, by entering a command like the following:
   ```
   cd <HANA_DVD>/DATA_UNITS/HANA_IM_LINUX__X86_64
   ```
3. Call the script in GUI mode to perform the rename:
   ```
   ./hanaconfig.sh --gui
   ```
4. Choose **HANA Distributed Setup**, and then **Next** to continue.
5. Choose **Add Additional Host** (or **Remove Additional Host**), and then choose **Next** to continue. When adding the additional host you need to specify what its role is: **worker** or **standby**.
6. Specify the required entries.
   For adding additional host in interactive mode, use the command `.hanaconfig.sh addhost` (or `removehost`).
   For adding additional host in batch mode, use the command `.hanaconfig.sh --batch addhost` (or `removehost`).

**Related Links**

*SAP HANA Installation Guide with Unified Installer Guide*

*SAP Note 611361– Host Names of SAP servers*

*Backing Up the SAP HANA Database [page 176]*

*About Persistent Data Storage in the SAP HANA Database [page 90]*

To ensure that the database can always be restored to its most recent committed state, changes to data in the database is periodically copied to disk. Logs containing data changes and certain transaction events are also saved regularly to disk. The data and logs of a system are stored in volumes.

*Redistributing Tables After Adding a Host [page 145]*

After you have added a new worker host to your SAP HANA system, you need to redistribute the tables in the system to balance the memory footprint of the tables and to improve performance (load balancing).
# 10.4 Call Options for Adding/Removing Hosts

## Table 12: Add Host Options

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Value</th>
<th>Mandatory</th>
<th>Default Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>addhost_sid &lt;SID&gt;</td>
<td>System ID</td>
<td>No</td>
<td>Detected</td>
<td>System ID of the new host</td>
</tr>
<tr>
<td>addhost_hostname &lt;hostname&gt;</td>
<td>Hostname</td>
<td>Yes</td>
<td></td>
<td>Hostname</td>
</tr>
<tr>
<td>addhost_master_password &lt;password&gt;</td>
<td>Password</td>
<td>Yes</td>
<td></td>
<td>System administrator password</td>
</tr>
<tr>
<td>addhost_root_user &lt;root user&gt;</td>
<td>Root user</td>
<td>Yes</td>
<td></td>
<td>Root user</td>
</tr>
<tr>
<td>addhost_root_user_password &lt;root user password&gt;</td>
<td>Root user password</td>
<td>Yes</td>
<td></td>
<td>Root user password</td>
</tr>
<tr>
<td>addhost_role &lt;role&gt;</td>
<td>Role</td>
<td>Yes</td>
<td></td>
<td>Additional host role</td>
</tr>
<tr>
<td>addhost_ha_group &lt;high availability group&gt;</td>
<td>High availability group</td>
<td>No</td>
<td></td>
<td>Failover group in High Availability scenarios</td>
</tr>
<tr>
<td>addhost_storage_partition</td>
<td>Storage partition</td>
<td>No</td>
<td></td>
<td>Storage partition number</td>
</tr>
</tbody>
</table>

## Table 13: Remove Host Options

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Value</th>
<th>Mandatory</th>
<th>Default Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>removehost_sid &lt;SID&gt;</td>
<td>System ID</td>
<td>No</td>
<td>Detected</td>
<td>System ID of the existing host</td>
</tr>
<tr>
<td>removehost_hostname &lt;hostname&gt;</td>
<td>Hostname</td>
<td>Yes</td>
<td></td>
<td>Hostname of the existing host</td>
</tr>
<tr>
<td>removehost_root_user &lt;root user&gt;</td>
<td>Root user</td>
<td>Yes</td>
<td></td>
<td>Root user of the existing host</td>
</tr>
<tr>
<td>removehost_root_user_password &lt;root user password&gt;</td>
<td>Root user password</td>
<td>Yes</td>
<td></td>
<td>Root user password of the existing host</td>
</tr>
</tbody>
</table>
10.5 Starting and Stopping a Distributed SAP HANA System

- You are logged on to the SAP system host as user <sid>adm or as a user with root permissions.
- You can use sapcontrol to start or stop all hosts in a scaled out system from the command line as follows:

**Starting the SAP System:**
You can start the SAP system by executing the following commands from the command line:

```
/usr/sap/hostctrl/exe/sapcontrol -nr <instance-nr> -function StartSystem
```

**Stopping the SAP System:**
You can stop the SAP system by executing the following commands from the command line:

```
/usr/sap/hostctrl/exe/sapcontrol -nr <instance-nr> -function StopSystem
```

**Current Host Status**
You can also query the current status of all hosts using the following commands from the command line:

```
/usr/sap/hostctrl/exe/sapcontrol -nr <instance-nr> -function GetSystemInstanceList
```

*Note:* HDB start or HDB stop only starts and stops the local host.

10.6 Table Distribution in SAP HANA

SAP HANA supports different ways of distributing data between multiple index servers in a single system:

- Different tables can be assigned to different index servers, which normally run on different hosts (database partitioning).
- A table can be split in a way that different rows of the table are stored on different index servers (table partitioning).

When a non-partitioned table is created in a distributed system, it must be assigned to one index server. By default, new tables are distributed across available index servers using a round-robin approach. For example, if there are three available index servers A, B, and C (including the master), the first table created will be located on server A, the next one on server B, the next on server C, and so on.

In addition, it is also possible to specify explicitly that a table or a partition be created on a specific index server using the following commands:

- ```CREATE TABLE <table_name> <table_contents_source> AT [LOCATION] 'host:port'```  
- ```CREATE TABLE <table_name> <table_contents_source> <partition_clause> AT [LOCATION] {'host:port' | {'host:port', ...}}```
Table Distribution for SAP NetWeaver Business Warehouse (BW) Mode

If you are running BW on SAP HANA and you want the table location to be decided by a predefined BW table distribution rule, the method parameter in the table_placement section of the `global.ini` file must be 2.

All the BW master data tables are distributed across the slave servers using round robin. Fact, DataStore Object (DSO), and Persistent Staging Area (PSA) tables are partitioned with the predefined rule and distributed to the slave servers. Thus, there are no BW data tables on the master server. All the row-store tables (ABAP system tables and general operation tables like REPOSRC) are created on the master server.

Note: This parameter is only applied when creating new tables.

This behavior is controlled by two additional parameters.

The parameter `bw_schema` in the table_placement section of the `global.ini` controls which schema is allowed on the master server. Tables in all other schemas are created on the slave servers. During installation this parameter is set to the BW schema and the BW shadow schema which is used during migration. Within the BW schema there are ABAP runtime tables and BW data tables. The BW data tables typically start with a namespace. Therefore a slash ("/") is used to separate these two kinds of data. All tables that start with a slash are always created on the slave servers. All other tables are only created on the master server if they are in one of the BW schemas.

The prefix parameter controls the naming convention, which is set to "/" by default.

Using Side-by-Side Replication within a SAP NetWeaver Business Warehouse (BW) Landscape

If you want to run a side-by-side replication scenario with tables replicated from a different system in the landscape there are two options to consider.

1. If the replicated tables are used for BW Virtual InfoProviders, it is advised to treat them like BW tables and locate them on slave servers (this is the default behavior).
2. Customers might want to have all replicated tables on the master server. In this case, it is required to add the schema name to the parameter `bw_schema` in the table_placement section of the `global.ini`.

Related Links

- **Monitoring Table Distribution** [page 148]
  To support the analysis and monitoring of performance issues in a distributed SAP HANA system, a table distribution editor is available in which you can see how tables are distributed across the hosts.
- **SAP Note 1637145 (SAP BW on HANA: Sizing SAP In-Memory Database)**
- **SAP HANA SQL Reference**
10.7 Table Partitioning in the SAP HANA Database

The partitioning feature of the SAP HANA database makes it possible to split column-store tables horizontally into disjunctive sub-tables or partitions. In this way, very large tables can be broken down into smaller, more manageable parts. Partitioning is typically used in distributed systems, but it may also be beneficial for single-host systems.

Partitioning is transparent for most SQL queries and data manipulation language (DML) statements. This means that there is no need to modify these statements to support partitioning.

When a table is partitioned, the split is done in such a way that each partition contains a different set of rows of the table. There are several alternatives available for specifying how the rows are assigned to the partitions of a table, for example, hash partitioning, partitioning by range or value. Partitioning a table based on multiple fields is also supported.

The following are the typical advantages of partitioning:

- **Load balancing in a distributed system**
  Individual partitions can be distributed across multiple hosts. This means that a query on a table is not processed by a single server but by all the servers that host partitions.

- **Overcoming the size limitation of column-store tables**
  A non-partitioned table cannot store more than 2 billion rows. It is possible to overcome this limit by distributing the rows across several partitions. Each partition must not contain more than 2 billion rows.

- **Parallelization**
  Partitioning allows operations to be parallelized by using several execution threads for each table.

- **Partition pruning**
  Queries are analyzed to determine whether or not they match the given partition specification of a table. If a match is found, it is possible to determine the actual partitions that hold the data being queried. Using this method, the overall load on the system can be reduced, thus improving the response time. For example, if a table is partitioned by year, a query restricted to the data of one year is executed only on the partition with data for this year.

- **Improved performance of the delta merge operation**
  The performance of the delta merge operation depends on the size of the main index. If data is only being modified on some partitions, fewer partitions will need to be delta merged and therefore performance will be better.

- **Explicit partition handling**
  Applications may actively control partitions, for example, by adding partitions to store the data for an upcoming month.

The following figure illustrates how a table can be distributed over three hosts with dedicated partitions for individual months.

**Figure 11: Example of Table Partitioning**
Write operations are only performed on the delta storage. In order to transform the data into a format that is optimized in terms of memory consumption and read performance, it must be transferred to the main storage. This is accomplished by the delta merge operation.

### 10.7.1 Single-Level Partitioning

When a table is partitioned, its rows are distributed to partitions according to different criteria known as partition specifications.

The SAP HANA database supports the following single-level partition specifications:

- Hash
- Range
- Round robin

For advanced use cases, these specifications can be nested using multi-level partitioning.

#### Hash Partitioning

Hash partitioning is used to distribute rows to partitions equally for load balancing and to overcome the 2 billion row limitation. The number of the assigned partition is computed by applying a hash function to the value of a specified field. Hash partitioning does not require an in-depth knowledge of the actual content of the table.

For each hash-partition specification, columns must be specified as partitioning columns. The actual values of these columns are used when the hash value is determined. If the table has a primary key, these partitioning columns must be part of the key. The advantage of this restriction is that a uniqueness check of the key can be performed on the local server. You can use as many partitioning columns as required to achieve a good variety of values for an equal distribution.

For more information about the SQL syntax for partitioning, see **SAP HANA SQL Reference**.

#### Example: Creating a Table with Hash Partitioning

<table>
<thead>
<tr>
<th>SQL Command</th>
<th>Result</th>
</tr>
</thead>
</table>
| CREATE COLUMN TABLE MY_TABLE (a INT, b INT, c INT, PRIMARY KEY (a, b)) PARTITION BY HASH (a, b) PARTITIONS 4 | - 4 partitions on columns a and b are created.  
- The target partition is determined based on the actual values in columns a and b. |

![Diagram of Host 1, Host 2, and Host 3 with months Jan, Feb, and Mar]
**SQL Command** | **Result**
---|---
● At least one column has to be specified.  
● If a table has a primary key, all partitioning columns must be part of that key.

```
CREATE COLUMN TABLE MY_TABLE (a INT, b INT, c INT, PRIMARY KEY (a,b)) PARTITION BY HASH (a, b) PARTITIONS
```

The number of partitions is determined by the database at runtime according to its configuration. It is recommended to use this function in scripts, and so on.

---

**Related Links**

*SAP HANA SQL Reference*

**Round-Robin Partitioning**

Round-robin partitioning is used to achieve an equal distribution of rows to partitions. However, unlike hash partitioning, you do not have to specify partitioning columns. With round-robin partitioning, new rows are assigned to partitions on a rotation basis.

Hash partitioning is usually more beneficial than round-robin partitioning for the following reasons:

- The partitioning columns can be evaluated in a pruning step. Therefore, all partitions are considered in searches and other database operations.
- Depending on the scenario, it is possible that the data within semantically-related tables resides on the same server. Some internal operations may then operate locally instead of retrieving data from a different server.

For more information about the SQL syntax for partitioning, see *SAP HANA SQL Reference*.

---

**Example: Creating a Table with Round-Robin Partitioning**

<table>
<thead>
<tr>
<th>SQL Command</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CREATE COLUMN TABLE MY_TABLE (a INT, b INT, c INT) PARTITION BY ROUNDROBIN PARTITIONS 4</code></td>
<td>4 partitions are created.</td>
</tr>
</tbody>
</table>

\*\*Note: The table must not have primary keys.\*

<table>
<thead>
<tr>
<th>SQL Command</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CREATE COLUMN TABLE MY_TABLE (a INT, b INT, c INT) PARTITION BY ROUNDROBIN PARTITIONS GET_NUM_SERVERS()</code></td>
<td>The number of partitions is determined by the database at runtime according to its configuration. It is recommended to use this function in scripts or clients that may operate in various landscapes.</td>
</tr>
</tbody>
</table>

---

**Related Links**

*SAP HANA SQL Reference*

**Range Partitioning**

Range partitioning can be used to create dedicated partitions for certain values or certain value ranges in table. Usually, this requires an in-depth knowledge of the values that are used or are valid for the chosen partitioning column. For example, a range partitioning scheme can be chosen to create one partition for each calendar month.
Applications may choose to use range partitioning to manage the partitioning of a table actively, that is, partitions may be created or dropped as needed. For example, an application may create a partition for an upcoming month so that new data is inserted into that new partition.

**Note:** Range partitioning is not well suited for load distribution. Multi-level partitioning specifications address this issue.

The range partition specification usually takes ranges of values to determine one partition, for example, 1 to 10. It is also possible to define a partition for a single value. In this way, a list partitioning known in other database systems can be emulated and combined with range partitioning.

When rows are inserted or modified, the target partition is determined by the defined ranges. If a value does not fit into one of these ranges, an error is raised. If this is not wanted, it is possible to define a rest partition where all rows that do not match any of the defined ranges are inserted. Rest partitions can be created or dropped on-the-fly as desired.

Range partitioning is similar to hash partitioning in that the partitioning column must be part of the primary key. Range partitioning is also restricted in terms of the data types that can be used. Only strings, integers, and dates are allowed.

For more information about the SQL syntax for partitioning, see *SAP HANA SQL Reference*.

### Example: Creating a Table with Range Partitioning

<table>
<thead>
<tr>
<th>SQL Command</th>
<th>Result</th>
</tr>
</thead>
</table>
| CREATE COLUMN TABLE MY_TABLE (a INT, b INT, c INT, PRIMARY KEY (a,b)) PARTITION BY RANGE (a) (PARTITION 1 <= VALUES < 5, PARTITION 5 <= VALUES < 20, PARTITION VALUE = 44, PARTITION OTHERS) | Partitions are created as follows:  
- 1 partition for values greater than or equal to 1 and less than 5  
- 1 partition for values greater than or equal to 5 and less than 20  
- 1 partition for values of 44  
- 1 rest partition is created for all values that creates partitions for ranges using <= VALUES < semantics  
- Partitioning column has to be part of the primary key  
- Only STRING, INT and DATE are allowed as data types for the partitioning column |

### Related Links

*SAP HANA SQL Reference*

### 10.7.2 Multi-Level Partitioning

Multi-level partitioning can be used to overcome the limitation of single-level hash partitioning and range partitioning, that is, the limitation of only being able to use key columns as partitioning columns. Multi-level partitioning makes it possible to partition by a column that is not part of the primary key.
For example, it may be desirable to leverage a date column to build partitions according to month or year. Multi-level partitioning is therefore the technical implementation of time-based partitioning.

The performance of the delta merge operation depends on the size of the main index of a table. If data is inserted into a table over time and it also contains temporal information in its structure, for example a date, multi-level partitioning may be an ideal candidate. If the partitions containing old data are infrequently modified, there is no need for a delta merge on these partitions: the delta merge is only required on new partitions where new data is inserted. Therefore, its run-time is constant over time as new partitions are being created and used.

The following figure shows how a hash partitioning at the first level and a range partitioning at the second level can be applied.

As mentioned above, there is a relaxation of the key column restriction. This applies only to the second level of partitioning (for hash-range and hash-hash).

When a row is inserted or updated, the unique constraint of the primary key must be checked. If the primary key has to be checked on all partitions across the landscape, this will involve expensive remote calls. Therefore, it is advantageous if only local partitions need to be checked. The concept of partition groups exists for this purpose. It allows inserts to occur whilst only requiring primary key checks on local partitions. All corresponding parts of the second level form a group.

In the figure above, parts 1 and 2 and parts 3 and 4 each form a group. When a row is inserted into part 1, it is only required to check for uniqueness on parts 1 and 2. All parts of a partition group must reside on the same host.

When using SQL commands to move partitions, be aware that it is not possible to move individual parts of partition groups, only partition groups as a whole.

**Hash-Range Partitioning**

Hash-range multi-level partitioning is the most common type of multi-level partitioning. Hash partitioning is implemented at the first level for load balancing and range partitioning at the second level for time-based partitioning.

The following figure shows a typical usage scenario. The load is distributed to three hosts using hash partitioning. Range partitioning is used at the second level to distribute the data to individual partitions according to month.
Example: Creating a Table with Hash-Range Multi-Level Partitioning

```
CREATE COLUMN TABLE MY_TABLE (a INT, b INT, c INT, PRIMARY KEY (a,b))
PARTITION BY
  HASH (a, b) PARTITIONS 4,
  RANGE (c)
  (PARTITION 1 <= VALUES < 5,
   PARTITION 5 <= VALUES < 20)
```

Related Links
SAP HANA SQL Reference

Round-Robin-Range Partitioning
Round-robin-range multi-level partitioning is the same as hash-range multi-level partitioning but with round-robin partitioning at the first level.

For more information about the SQL syntax for partitioning, see SAP HANA SQL Reference.

Example: Creating a Table with Round-Robin-Range Partitioning

```
CREATE COLUMN TABLE MY_TABLE (a INT, b INT, c INT)
PARTITION BY
  ROUNDROBIN PARTITIONS 4,
  RANGE (c)
  (PARTITION 1 <= VALUES < 5,
   PARTITION 5 <= VALUES < 20)
```

Related Links
SAP HANA SQL Reference

Hash-Hash Partitioning
Hash-hash multi-level partitioning is implemented with hash partitioning at both levels. The advantage of this is that the hash partitioning at the second level may be defined on a non-key column.

For more information about the SQL syntax for partitioning, see SAP HANA SQL Reference.
Example: Creating a Table with Hash-Hash Partitioning

CREATE COLUMN TABLE MY_TABLE (a INT, b INT, c INT, PRIMARY KEY (a,b))
PARTITION BY
HASH (a, b) PARTITIONS 4,
HASH (c) PARTITIONS 7

Related Links
SAP HANA SQL Reference

10.7.3 Using Date Functions to Partition
Use the date functions to partition a table’s date or timestamp column by year or month.

If a table needs to be partitioned by month or by year and it contains only a date column or a timestamp column, you can use the date functions to restrict your query results by year or by year and month.

Example: Creating a Table with Hash Partitioning and Date Function

<table>
<thead>
<tr>
<th>SQL Command</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATE COLUMN TABLE MY_TABLE (a DATE, b INT, PRIMARY KEY (a,b)) PARTITION BY HASH (year(a)) PARTITIONS 4</td>
<td>If a value takes the format “2012-01-08”, the hash function is only applied to “2012”. This function can also be used for pruning.</td>
</tr>
<tr>
<td>CREATE COLUMN TABLE MY_TABLE (a DATE, b INT, PRIMARY KEY (a,b)) PARTITION BY RANGE (year(a)) (PARTITION ‘2005’ &lt;= values &lt; ‘2008’, PARTITION ‘2008’ &lt;= values &lt; ‘2011’)</td>
<td>Partition by range using the year.</td>
</tr>
<tr>
<td>CREATE COLUMN TABLE MY_TABLE (a DATE, b INT, PRIMARY KEY (a,b)) PARTITION BY RANGE (month(a)) (PARTITION '2005-01' &lt;= values &lt; '2005-07', PARTITION '2005-07' &lt;= values &lt; '2006-01')</td>
<td>Partition by range using the year and month.</td>
</tr>
</tbody>
</table>

10.7.4 Explicit Partition Handling for Range Partitioning
For all partition specifications involving range, it is possible to have additional ranges added and removed as necessary. This means that partitions are created and dropped as required by the ranges in use. In the case of multi-level partitioning, the desired operation is applied to all nodes.

Note: If a partition is created and a rest partition exists, the rows in the rest partition that match the newly-added range are moved to the new partition. If the rest partition is large, this operation may take a long time. If a rest partition does not exist, this operation is fast as only a new partition is added to the catalog.

Range partitioning requires at least one range to be specified regardless of whether or not there is a rest partition. When partitions are dropped, the last partition created cannot be dropped even if a rest partition exists.

Caution: The DROP PARTITION command deletes data. It does not move data to the rest partition.
For more information about the SQL syntax for partitioning, see SAP HANA SQL Reference.

**Example: Changing a Table to Add/Drop Partitions**

```sql
ALTER TABLE MY_TABLE ADD PARTITION 100 <= VALUES < 200
ALTER TABLE MY_TABLE DROP PARTITION 100 <= VALUES < 200
```

**Example: Changing a Table to Add/Drop Rest Partition**

```sql
ALTER TABLE MY_TABLE ADD PARTITION OTHERS
ALTER TABLE MY_TABLE DROP PARTITION OTHERS
```

**Related Links**

SAP HANA SQL Reference

### 10.7.5 Changing Partitioning

How a table is partitioned can be determined upon creation or at a later point in time. There are several ways to do this:

- Change a non-partitioned table into a partitioned table and the other way round
- Change the partitioning of a table as follows:
  - Change the partitioning specification, for example, from hash to round-robin
  - Change the partitioning column
  - Split partitions
  - Merge partitions

Changing a table’s partitioning in the above ways can be costly for the following reasons:

- It takes a long time to run, up to several hours for huge tables.
- It has relatively high memory consumption.
- It requires an exclusive lock (only selects are allowed).
- It performs a delta merge in advance.
- It writes everything to the log (required for backup and recovery).

It is recommended that you (re-)partition tables before inserting mass data or while they are still small. If a table is not partitioned and its size reaches configurable absolute thresholds, or if a table grows by a certain percentage per day, the statistics server issues an alert.

There are three types of re-partitioning:

- From n to m partitions, where m is not a multiple/divider of n, for example, from HASH 3 X to HASH 2 X:
From n to n partitions using a different partition specification or different partitioning columns, for example, HASH 3 X to HASH 3 Y:

From n to m partitions, where m is a multiple/divider of n, for example, HASH 3 X to HASH 6 X:

In the first two cases, all source parts must be located on the same host. Up to one thread per column is used to split/merge the table. In the third case, it is not necessary to move all parts to the same host. Instead, the split/merge request is broadcast to each host where partitions reside. Up to one thread per column and source part is used. This type of split/merge operation is typically faster as it is always recommended to choose a multiple or divider of the source parts as number of target parts. This type of re-partitioning is called a parallel split/merge.

Example: Partitioning/Merging a Table

<table>
<thead>
<tr>
<th>SQL Command</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER TABLE MY_TABLE PARTITION BY ...</td>
<td>This can be applied to non-partitioned tables and to partitioned tables.</td>
</tr>
<tr>
<td>SQL Command</td>
<td>Note</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>ALTER TABLE MY_TABLE MERGE PARTITIONS</td>
<td>• Depending on the type of the split/merge operation (see above) it may be necessary to move partitions beforehand.</td>
</tr>
<tr>
<td></td>
<td>• All parts of the partitioned table are merged into a non-partitioned table.</td>
</tr>
<tr>
<td></td>
<td>• All source partitions must reside on the same server.</td>
</tr>
</tbody>
</table>

For more information about the SQL syntax for (re-)partitioning and merging, see SAP HANA SQL Reference.

Note: Although it is possible to re-partition tables and merge partitions manually, in some situations it may be more effective to use the redistribution operation available for optimizing table partitioning (for example, if a change of partition specification is not required). Redistribution operations use complex algorithms to evaluate the current distribution and determine a better distribution depending on the situation.

Parallelism and Memory Consumption

Partitioning and merge operations consume a high amount of memory. To reduce the memory consumption, it is possible to configure the number of threads used. You can change the default value of the parameter `split_threads` in the partitioning section of the `indexserver.ini` configuration file. By default, 16 threads are used. In the case of a parallel partition/merge, the individual operations use a total of the configured number of threads for each host. Each operation takes at least one thread.

If a table has a history index, it is possible to partition the main and history index in parallel. Set the parameter `split_history_parallel` in the section partitioning in the `indexserver.ini` configuration. The default value is no.

Related Links

- SAP HANA SQL Reference
- Optimizing Table Partitioning [page 147]
- The Delta Merge Operation [page 107]

In a distributed SAP HANA system, partitioned tables are distributed across different index servers. The location of the different partitions can be specified manually or determined by the database when the table is initially partitioned. Over time, this initial partitioning may no longer be optimal, for example, if a partition has grown significantly.

Write operations are only performed on the delta storage. In order to transform the data into a format that is optimized in terms of memory consumption and read performance, it must be transferred to the main storage. This is accomplished by the delta merge operation.
10.8 Redistribution of Tables in a Distributed SAP HANA System

In a distributed SAP HANA system, tables and table partitions are assigned to an index server on a particular host at their time of creation, but this assignment can be changed. In certain situations, it is even necessary.

For example, if you plan to remove a host from your system, then you first need to move all the data on that host first to the other hosts in the system. Redistributing tables may also be useful if you suspect that the current distribution is no longer optimal.

Although it is possible to move tables and table partitions manually from one host to another, this is neither practical nor feasible for a large-scale redistribution of data. SAP HANA supports several redistribution operations that use complex algorithms to evaluate the current distribution and determine a better distribution depending on the situation.

Redistribution operations are available to support the following situations:

- You are planning to remove a host from your system
- You have added a new host to your system
- You want to optimize current table distribution
- You want to optimize table partitioning

Note: For OLTP scenarios, other routing rules must be applied for the redistribution operation. This is described in SAP Note 1781986 (Business Suite on SAP HANA Scale Out).

Related Links

Monitoring Table Distribution [page 148]
To support the analysis and monitoring of performance issues in a distributed SAP HANA system, a table distribution editor is available in which you can see how tables are distributed across the hosts.

About Scalability [page 127]
SAP Note 1781986

10.8.1 Saving Current Table Distribution

Changing how tables are distributed across the hosts of a distributed SAP HANA system is a critical operation. Therefore, before executing a redistribution operation, it is recommended that you save the current distribution of table data so that it can be restored if necessary.

To be able to save the current table distribution, you must have the system privilege RESOURCE ADMIN and at least the object privilege ALTER for all schemas involved.

1. In the Administration editor, choose Landscape & Redistribution.
2. Choose Save.
   The Table Redistribution dialog box appears.
3. Choose Next.
   The system generates a redistribution plan that shows the distribution that will be saved.

Note: Saving the current table operation involves the execution of a redistribution operation even though an actual redistribution of data does not take place.
4. Choose **Execute**.
   The system saves the current table distribution. The associated redistribution operation appears in the list of executed operations.

You can now restore the saved table distribution at later point in time if necessary.

### 10.8.2 Redistributing Tables Before Removing a Host

Before you can remove a host from your SAP HANA system, you must move the tables on the index server of the host in question to the index servers on the remaining hosts in the system.

To be able redistribute tables across the hosts in your system, you must have the system privilege **RESOURCE ADMIN** and at least the object privilege **ALTER** for all schemas involved. As redistributing data is a critical operation, it is also recommended that you have saved the current distribution so you can restore it if necessary.

1. In the Administration editor, choose **Landscape** ➔ **Configuration** ➔ **Redistribution**.
2. From the context menu of the host that you plan to remove, choose **Remove Host...**
3. In the **Remove Host** dialog box, choose **Yes**.
   The system marks the host for removal and executes the required redistribution operation. This results in the data on the index server of the host being moved to the index servers of the remaining hosts in the system.

   The redistribution operation appears in the list of executed operations on the **Redistribution** tab.

You can remove the host.

![Caution:](image)

**Caution:** After you remove the host from your system, you must perform a data backup to ensure that you can recover the database to a point in time after you removed the host.

**Related Links**

- [Backing Up the SAP HANA Database](page 176)
- [Perform a Distributed SAP HANA Setup](page 130)

### 10.8.3 Redistributing Tables After Adding a Host

After you have added a new worker host to your SAP HANA system, you need to redistribute the tables in the system to balance the memory footprint of the tables and to improve performance (load balancing).

To be able redistribute tables across the hosts in your system, you must have the system privilege **RESOURCE ADMIN** and at least the object privilege **ALTER** for all schemas involved. As redistributing data is a critical operation, it is also recommended that you have saved the current distribution so you can restore it if necessary.

1. In the Administration editor, choose **Landscape** ➔ **Configuration** ➔ **Redistribution**.
2. In the **Redistribution Operations** area, select **Redistribute tables after adding host(s)** and choose **Execute**.
   The **Table Redistribution** dialog box appears.
3. Choose **Next**.
   The system evaluates the current distribution of tables and generates a redistribution plan. This plan specifies which tables will be moved where.
4. Review the redistribution plan to ensure that you want to proceed and choose **Execute**.
   The system redistributes the tables in your system across all available index servers. The associated redistribution operation appears in the list of executed operations.
Caution: After you add a host to your system, you must perform a data backup to ensure that you can recover the database to a point in time after you added the host.

Related Links
- Backing Up the SAP HANA Database [page 176]
- Perform a Distributed SAP HANA Setup [page 130]

10.8.4 Restoring Previous Table Distribution
Changing how tables are distributed across the hosts of an SAP HANA system is a critical operation. You may need to restore the table distribution from a previous point in time.

To be able to restore a previous table distribution, you must have the system privilege RESOURCE ADMIN and at least the object privilege ALTER for all schemas involved.

1. In the Administration editor, choose Landscape \ Configuration \ Redistribution.
2. In the Executed Operations area, identify the operation that corresponds to the table distribution that you want to restore.
   For example, you saved the table distribution at a particular point in time and you want to revert to this configuration.
3. Check the redistribution plan of the operation to ensure that you want to proceed.
   To do this, select the operation and choose Show Plan...
4. Select the operation and choose Restore.
The system restores the selected table distribution.
The associated redistribution operation appears in the list of executed operations.

10.8.5 Optimizing Table Distribution
During productive operation, you may discover that the initial assignment of tables and table partitions to index servers is no longer optimal, for example, frequently joined tables are located on different servers. You can therefore trigger a redistribution operation that evaluates the current situation and determines how distribution can be improved.

To be able to optimize table distribution, you must have the system privilege RESOURCE ADMIN and at least the object privilege ALTER for all schemas involved.

1. In the Administration editor, choose Landscape \ Configuration \ Redistribution.
2. In the Redistribution Operations area, select Optimize table distribution and choose Execute.
The Table Redistribution dialog box appears.
3. Choose Next.
The system evaluates the current distribution of tables and partitions, and whether or not partitioned tables need to be repartitioned. A redistribution plan is subsequently generated, specifying which tables and partitions will be moved where, and how partitioned tables will be repartitioned and new partitions distributed.
4. Review the redistribution plan to ensure that you want to proceed and choose Execute.
The system redistributes and repartitions the tables and partitions in your system.
10.8.6 Optimizing Table Partitioning

In a distributed SAP HANA system, partitioned tables are distributed across different index servers. The location of the different partitions can be specified manually or determined by the database when the table is initially partitioned. Over time, this initial partitioning may no longer be optimal, for example, if a partition has grown significantly.

To be able optimize partitioning, you must have the system privilege RESOURCE ADMIN and at least the object privilege ALTER for all schemas involved. As redistributing data is a critical operation, it is also recommended that you have saved the current distribution so you can restore it if necessary.

1. In the Administration editor, choose ❯ Landscape ❯ Redistribution ❯
2. In the Redistribution Operations area, select Optimize table partitioning and choose Execute. The Table Redistribution dialog box appears.
3. Choose Next.
4. Review the redistribution plan to ensure that you want to proceed and choose Execute. The system re-partitions the required tables and distributes the new partitions in your system.

Note: Partitioning is a potentially expensive operation both in terms of time and memory consumption.

The associated redistribution operation appears in the list of executed operations.

10.8.7 Modifying Table Distribution Manually

In a distributed SAP HANA system, you can move individual tables or table partitions from the index server of one host to the index server of another.

To move a table or table partition from one host to another, you must have system privilege DATA ADMIN, or system privilege CATALOG READ and SQL privilege ALTER for the table being moved. If you have system privilege CATALOG READ and you are the owner of the table, then you can also move the table without SQL privilege ALTER.
Note: For a large-scale redistribution of data, it is recommended that you execute one of the available redistribution operations instead of modifying table distribution manually as described here. Redistribution operations use complex algorithms to evaluate the current distribution and determine a better distribution depending on the situation.

1. Open the table distribution editor by right-clicking the required entry in the navigator and then choosing **Show Table Distribution**:
   - Catalog
   - Schema
   - Table

Note: For performance reasons, not all tables of the selected schema are displayed, but only the first 1000 tables. You can change this setting in preferences of the Administration Console (Windows Preferences). If more tables exist in the selected schema, a message is displayed.

2. If necessary, use the filtering options to refine the list of tables displayed. For example, you can restrict the display to tables on specific hosts only.

3. To view the detailed distribution information of a partitioned table, select the table in the overview list. The **Table Partition Details** area is then shown underneath.

Note: You can only see table partition information for tables of type COLUMN as this is the only table type that can be partitioned.

4. To move a table to another host, proceed as follows:
   a) Right-click the table in the overview list and choose **Move Table...**
   b) Specify the host to which you want to move the table.

5. To move a table partition or sub-partition to another host, proceed as follows:
   a) Right-click the partition or sub-partition in the **Table Partition Details** area and choose **Move Partitions...**
   b) Specify the host to which you want to move the partition(s).

It is not possible to split or merge tables in either the table distribution editor or the table editor of the SAP HANA studio. You must perform these operations either using SQL statements or through the execution of a redistribution operation (for example, **Optimize table distribution** or **Optimize table partitioning**).

Related Links
*Table Partitioning in the SAP HANA Database* [page 134]

The partitioning feature of the SAP HANA database makes it possible to split column-store tables horizontally into disjunctive sub-tables or partitions. In this way, very large tables can be broken down into smaller, more manageable parts. Partitioning is typically used in distributed systems, but it may also be beneficial for single-host systems.

### 10.9 Monitoring Table Distribution

To support the analysis and monitoring of performance issues in a distributed SAP HANA system, a table distribution editor is available in which you can see how tables are distributed across the hosts.
In the case of partitioned tables, you can also see how the individual partitions and sub-partitions are distributed, as well as detailed information about the physical distribution, for example, part ID, partition size, and so on.

**Note:** You can see the detailed distribution information of an individual table by viewing its table definition (Runtime Information tab).

1. Open the **Table Distribution** editor by right-clicking any of the following entries in the navigator and then choosing **Show Table Distribution**:
   - Catalog
   - Schema
   - Table

   **Note:** For performance reasons, not all tables of the selected schema are displayed, but only the first 1,000 tables. You can change this setting in the preferences of the Administration Console. If more tables exist in the selected schema, a message is displayed.

2. If necessary, use the filtering options to refine the list of tables displayed. For example, you can restrict the display to tables on specific hosts only.

3. To view the detailed distribution information of a partitioned table, select the table in the overview list. The **Table Partition Details** area is then shown underneath.

   **Note:** You can only see table partition information for column-store tables as this is the only table type that can be partitioned.

**Related Links**

*Table Partitioning in the SAP HANA Database [page 134]*

The partitioning feature of the SAP HANA database makes it possible to split column-store tables horizontally into disjunctive sub-tables or partitions. In this way, very large tables can be broken down into smaller, more manageable parts. Partitioning is typically used in distributed systems, but it may also be beneficial for single-host systems.

*Redistribution of Tables in a Distributed SAP HANA System [page 144]*

In a distributed SAP HANA system, tables and table partitions are assigned to an index server on a particular host at their time of creation, but this assignment can be changed. In certain situations, it is even necessary.

*Customizing the Administration Console [page 19]*

There are many options available for customizing the Administration Console of the SAP HANA studio.

### 10.10 System Migration

You can migrate or copy the contents of your database from a source system to a new target system. The following scenarios are supported:

- Source and target systems are both SAP HANA systems and have the same number of hosts
- Source and target systems are both SAP HANA systems and have a different number of hosts
- The source system is not an SAP HANA system
10.10.1 Migrating from One SAP HANA System to Another (Same Number of Hosts)

1. Back up the source system.
   Execute the command: `BACKUP DATA ALL USING FILE('<path><prefix>')`

   **Note:** Set the shared disk storage of the target system as the target backup location to avoid any additional file copying between two different storage locations.

2. Install the target system with the same number of hosts.
3. Recover to the target system.
4. Adjust the userstore, if the SID of the source ABAP system is different from the SID of the target ABAP system.

   If you recovered the system from a backup and the SID of the source ABAP system is different to the SID of the target ABAP system, you should adjust your user privileges. You can do this by executing the following command from your appserver account.

   ```
   $ hdbuserstore SET DEFAULT localhost:38215 SYSTEM manager
   $ hdbuserstore SET DEFAULT localhost:38215 YOURSCHEMANAME YOURPASSWORD
   ```

   **Note:**
   YOURSCHEMANAME and YOURPASSWORD are the schema name and password used in your backed-up system. In order to use `hdbuserstore`, you should install it from the SAP HANA installation by executing the command `./hdbinst -a client`.

5. Carry out post-migration operations.

**Related Links**
- [Backing Up the SAP HANA Database](#) [page 176]
- [Perform a Distributed SAP HANA Setup](#) [page 130]
- [Redistributing Tables After Adding a Host](#) [page 145]

After you have added a new worker host to your SAP HANA system, you need to redistribute the tables in the system to balance the memory footprint of the tables and to improve performance (load balancing).

10.10.2 Migrating from One SAP HANA System to Another (Different Number of Hosts)

1. Back up the source system.
   Execute the command: `BACKUP DATA ALL USING FILE('<path><prefix>')`

   **Note:** Set the shared disk storage of the target system as the target backup location to avoid any additional file copying between two different storage locations.

2. Install the target system with the same number of hosts as the source system.
   For example, if your source system has three hosts and the target system has five, you need to install the target system initially with three hosts. Otherwise, migration is not possible.

3. Recover to the target system.
4. Add further hosts to the target system as necessary.
   Following the example above, you now have to add two additional nodes to the target system.

5. Adjust the userstore, if the SID of the source ABAP system is different from the SID of the target ABAP system.
   If you recovered the system from a backup and the SID of the source ABAP system is different to the SID of the target ABAP system, you should adjust your user privileges. You can do this by executing the following command from your appserver account:

   ```
   $ hdbuserstore SET DEFAULT localhost:38215 SYSTEM manager
   $ hdbuserstore SET DEFAULT localhost:38215 YOURSCHEMANAME YOURPASSWORD
   ```

   Note:
   YOURSCHEMANAME and YOURPASSWORD are the schema name and password used in your backed-up system. In order to use hdbuserstore, you should install it from the SAP HANA installation by executing the command: ./hdbinst -a client

6. Perform a full data backup.
   Execute the command: BACKUP DATA ALL USING FILE ('<filename>')</p>

7. Carry out post-migration operations.

8. Execute the table redistribution operation `Redistribute tables after adding host(s)`.

Related Links
- **Backing Up the SAP HANA Database** [page 176]
- **Perform a Distributed SAP HANA Setup** [page 130]
- **Redistributing Tables After Adding a Host** [page 145]

After you have added a new worker host to your SAP HANA system, you need to redistribute the tables in the system to balance the memory footprint of the tables and to improve performance (load balancing).

### 10.10.3 Migrating from a Non-SAP HANA System

1. Export the source system.
2. Install and start the target system with the desired number of hosts.
3. Import the exported data into the target system.
4. Execute post-migration operations.
5. Execute the table redistribution operation `Redistribute tables after adding host(s)`.

Related Links
- **Backing Up the SAP HANA Database** [page 176]
- **Perform a Distributed SAP HANA Setup** [page 130]
- **Redistributing Tables After Adding a Host** [page 145]

After you have added a new worker host to your SAP HANA system, you need to redistribute the tables in the system to balance the memory footprint of the tables and to improve performance (load balancing).
11 Managing Encryption of Data Volumes in the SAP HANA Database

11.1 Data Volume Encryption

To ensure that your database can always be restored to its most recent committed state, all data in the database is periodically copied to disk. You can ensure the privacy of data on disk by enabling persistence encryption.

Ideally, you enable encryption immediately after installation or upgrade of SAP HANA. However, if you are already operating an SAP HANA database, it is also possible to enable persistence encryption and have your data encrypted retroactively.

Related Links
SAP HANA Installation Guide with Unified Installer
SAP HANA Security Guide

11.2 Enabling Persistence Encryption in an Existing SAP HANA System

There are two ways to enable persistence encryption in an existing SAP HANA system. The recommended way involves reinstalling your system. If this is not possible (for example, because it would result in too much downtime), you can enable encryption immediately.

However, the resulting protection may be incomplete because, due to the shadow memory nature of SAP HANA persistence, outdated versions of pages may still remain unencrypted on disk. This method is therefore not recommended.

Note: To enable persistence encryption and access the encryption monitoring views, you must have the system privilege RESOURCE ADMIN.

<table>
<thead>
<tr>
<th>Method</th>
<th>Procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reinstallation</td>
<td>1. Perform a data backup.</td>
</tr>
<tr>
<td>(Recommended)</td>
<td>2. Uninstall your system.</td>
</tr>
<tr>
<td></td>
<td>If possible, overwrite the former data area with random values.</td>
</tr>
<tr>
<td></td>
<td>3. Reinstall your system.</td>
</tr>
</tbody>
</table>
Method | Procedure
--- | ---
4. | Enable persistence encryption by executing the following SQL command, for example in the SQL console:

```
ALTER SYSTEM PERSISTENCE ENCRYPTION ON
```
5. | Recover your system.

All pages in the data area on disk are now encrypted and encryption is active. You can verify that this is the case in the monitoring view `M_PERSISTENCE_ENCRYPTION_STATUS`. The column `IS_ENCRYPTION_ACTIVE` contains the value `TRUE`.

When changes to data are persisted to disk, the relevant pages are automatically encrypted as part of the write operation.

Immediate enabling (not recommended)

1. | Log on to the SAP HANA system host as the operating system user, `<sid>adm`.
2. | Generate a root encryption key from the command line using the `hdbnsutil` program.

⚠️ **Caution:** Do not generate the root encryption key after you have enabled persistence encryption, since this would render the SAP HANA database unusable.

The command is as follows:

```
#hdbnsutil -generateRootKeys --type=ALL
```
3. | Enable persistence encryption by executing the following SQL command, for example in the SQL console:

```
ALTER SYSTEM PERSISTENCE ENCRYPTION ON
```

Encryption is now active for new data saved to disk as of the next savepoint operation.

**Note:** The frequency at which savepoints are defined can be configured in the persistence section of the `global.ini` file (every 5 minutes by default). Savepoints are also triggered automatically by a number of other operations such as data backup, and database shutdown and restart. You can trigger a savepoint manually by executing the following statement

```
ALTER SYSTEM SAVEPOINT
```

You can verify that encryption is active in the monitoring view `M_PERSISTENCE_ENCRYPTION_STATUS`. The column `ENCRYPTION_ACTIVE_AFTER_NEXT_SAVEPOINT` contains the value `TRUE`. 
Method | Procedure
---|---
In addition, existing data is encrypted in the background. Depending on the size of the SAP HANA system, this process can be very time consuming. Only after this process has completed is all your data encrypted. You can verify that this is the case in the monitoring view `M_PERSISTENCE_ENCRYPTION_STATUS`. The column `DATA_CONVERSION_ACTIVE` contains the value `TRUE`.

Even then, due to the shadow memory nature of SAP HANA database persistence, the data area may still contain outdated, unencrypted versions of pages. This approach is therefore not recommended.

**Note:** You can disable persistence encryption by executing the following SQL command `ALTER SYSTEM PERSISTENCE ENCRYPTION OFF`. After the next savepoint operation, encryption is no longer active and decryption of encrypted data starts in the background.

**Related Links**
- *Back up the SAP HANA Database* [page 176]
- *Recover the SAP HANA Database* [page 181]
- *SAP HANA Security Guide*
- *SAP HANA Installation Guide with Unified Installer*
- *SAP HANA Database - System Tables and Monitoring Views*

### 11.3 Changing Page Encryption Keys

If you are using the SAP HANA persistence encryption feature, we recommend that you periodically change the encryption key used to encrypt pages in the data area in line with your organization’s security policy.

To change page encryption keys and access the encryption monitoring views, you must have the system privilege `RESOURCE ADMIN`.

You can see how long the current encryption key has been valid in the monitoring view `M_PERSISTENCE_ENCRYPTION_KEYS`.

1. Open the SQL console.
2. Execute the following command:
   ```sql
   ALTER SYSTEM PERSISTENCE ENCRYPTION CREATE NEW KEY
   ```

After the next savepoint operation, a new random encryption key is generated. This key will be used to encrypt pages as of the next savepoint operation. Depending on the workload of the database, this may not happen for some time. Pages that were previously written to disk are not re-encrypted.

**Note:** The frequency at which savepoints are defined can be configured in the persistence section of the `global.ini` file (every 5 minutes by default). Savepoints are also triggered automatically by a
You can verify this result in the monitoring view M_PERSISTENCE_ENCRYPTION_STATUS. The column KEY_CHANGE_WITH_NEXT_SAVEPOINT contains the value TRUE.

**Note:** Encryption keys that are no longer in use are automatically removed the next time the database is restarted.

### Related Links
- SAP HANA Database - System Tables and Monitoring Views

## 11.4 Re-Encrypting the Entire Data Area

If you are using the SAP HANA persistence encryption feature, you may need to re-encrypt your entire data area for different reasons. For example, you have a lot of encryption keys in your system, an encryption key was compromised, or your organization’s security policy requires that all data be encrypted with keys not older than a certain age.

You can see all encryption keys in your system and their validity periods in the monitoring view M_PERSISTENCE_ENCRYPTION_KEYS.

**Note:** To re-encrypt the data area and access the encryption monitoring views, you must have the system privilege RESOURCE ADMIN.

1. Open the SQL console.
2. Execute the following command:

   ```sql
   ALTER SYSTEM PERSISTENCE ENCRYPTION APPLY CURRENT KEY
   ```

   After the next savepoint operation, pages that have been encrypted with an old key are decrypted using the old key and re-encrypted with the current key.

**Note:** The frequency at which savepoints are defined can be configured in the persistence section of the global.ini file (every 5 minutes by default). Savepoints are also triggered automatically by a number of other operations such as data backup, and database shutdown and restart. You can trigger a savepoint manually by executing the following statement ALTER SYSTEM SAVEPOINT.

**Note:** If you created a new key by executing the SQL command ALTER SYSTEM PERSISTENCE ENCRYPTION CREATE NEW KEY before executing ALTER SYSTEM PERSISTENCE ENCRYPTION APPLY CURRENT KEY, this new key is used for re-encryption.

### Related Links
- SAP HANA Database - System Tables and Monitoring Views
12 Auditing Activity in SAP HANA Systems

12.1 Activating and Configuring Auditing for an SAP HANA System

The auditing feature of the SAP HANA database allows you to monitor and record selected actions performed in your system. To be able to use this feature, it must first be activated for the system. It is then possible to create and activate the required audit policies.

To be able to activate and configure auditing for an SAP HANA system, you must have the system privilege AUDIT ADMIN or INIFILE ADMIN.

1. In the Navigator view, open the Security editor of the system to be audited.
2. In the System Settings for Auditing area, set the auditing status to Enabled.
3. Optional: Configure the target of the audit trail.
   - The logging system of the Linux operating system (syslog) is the only audit trail target supported for productive systems. However, for test purposes in non-productive systems, you can change the audit trail type to CSVTEXTFILE and specify a target directory. If you leave the target directory empty, the file is written to the same directory as trace files (for example, indexserver_<host.<port.audit_trail.csv).
   - Caution: You must not use CSVTEXTFILE for a productive system as it has severe restrictions. Firstly, it is not sufficiently secure. By default, this file is written to the same directory as trace files, so database users with the system privilege DATA ADMIN, CATALOG READ, TRACE ADMIN, or INIFILE ADMIN can access it. At operating system level, any user in the SAPSYS group can access it. Secondly, audit trails are created for each server in a distributed system. This makes it more difficult to trace audit events that were executed across multiple servers (distributed execution).
4. Choose the (Deploy) button.

Auditing is now activated in your system and you can create the required audit policies.

Related Links
- SAP HANA Security Guide
- SAP HANA SQL Reference

12.2 Creating an Audit Policy

Auditing is implemented through the creation and activation of audit polices. An audit policy defines the actions to be audited, as well as the conditions under which the action must be performed to be relevant for auditing.

To be able to create an audit policy, you must have the system privilege AUDIT ADMIN.
1. In the **Navigator** view, open the Security editor of the system to be audited.

2. In the **Audit Policies** area, choose **Create New Policy**.
   A new line is added to the list of policies.

3. Enter the policy name.
   The policy name can contain only letters (Aa-Zz), numbers (0-9), and underscores (_).

4. Specify the actions to be audited as follows:
   a) In the **Audited Actions** column, choose the *** button.
      The **Edit Actions Audited by <policy_name>** dialog box appears.
   b) Select the required actions.
   c) Choose **OK**.

5. Specify the action status.
   The action status specifies when the actions in the policy are to be audited. The following values are possible:

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUCCESSFUL</td>
<td>The action is audited only when the SQL statement is successfully executed.</td>
</tr>
<tr>
<td>UNSUCCESSFUL</td>
<td>The action is audited only when the SQL statement is unsuccessfully executed.</td>
</tr>
<tr>
<td>ALL</td>
<td>The action is audited when the SQL statement is both successfully and</td>
</tr>
<tr>
<td></td>
<td>unsuccessfully executed.</td>
</tr>
</tbody>
</table>

   **Note:** An unsuccessful attempt to execute an action means that the user was not authorized to execute the action. If another error occurs (for example, misspellings in user or object names and syntax errors), the action is generally not audited. In the case of actions that involve data manipulation (that is, INSERT, SELECT, UPDATE, DELETE, and EXECUTE statements), additional errors (for example, invalidate views) are audited.

6. Specify the audit level.
   The audit level specifies the severity of the audit entry written to the audit trail when the actions in the policy occur.

7. If necessary, specify the user(s) to be audited.
   The actions in the policy will only be audited when performed by the specified user(s). If you do not specify a user, the actions will be audited regardless of who performs them.

   **Note:** You must specify a user if you chose to audit all auditable actions.

8. If necessary, specify the target object(s) to be audited.
   You must specify a target object if the actions to be audited involve data manipulation, for example, the actions SELECT, INSERT, UPDATE, DELETE, and EXECUTE. The actions in the policy will only be audited when they are performed on the specified object or objects.

   When specifying target objects, note the following:
   ○ You can only enter tables, views, and procedures.
The target object must be valid for all actions in the policy.
You can only enter objects that exist. However, if the object is deleted, the audit policy remains valid. This means that if the object is recreated, that is the same object type with the same name is created, the audit policy will work for this object again.

9. Choose the Deploy button.

The list of audit policies is saved together with the new policy. The new policy is automatically enabled. This means that when an action in the policy now occurs under the conditions defined in the policy, an audit entry is created in the audit trail. You can disable a policy at any time by changing the policy status. It is also possible to delete a policy.

Note: Audit policies are not owned by the database user who creates them and therefore will not be deleted if the corresponding database user is deleted.

Related Links
SAP HANA Security Guide
SAP HANA SQL Reference
13   Backing Up and Recovering the SAP HANA Database

13.1   SAP HANA Database Backup and Recovery

To maintain optimal performance, the SAP HANA database holds the bulk of its data in memory. However, SAP HANA uses persistent storage to provide a fallback in the event of a failure.

During normal database operation, data is automatically saved from memory to disk at regular savepoints. Additionally, all data changes are recorded in the redo log. With each committed database transaction, the redo log is saved from memory to disk. If a failure occurs, for example a power outage, the database can be restarted in the same way as any disk-based database, and it is returned to its last consistent state by replaying the redo log since the last savepoint.

While savepoints and log writing protect your data against power outages, savepoints do not help if the persistent storage itself is corrupted. To protect against data loss resulting from hardware failure, backups are required. Backups save the payload (that is, the actual data) in the data area and log area to different backup destinations.

Backups are performed while the database is running. While a backup is running, the impact on system performance is negligible, and users can continue to work normally.

About this Backup and Recovery Documentation

This documentation describes how to back up and recover SAP HANA database.

The following tasks are described:

- Create a data backup of SAP HANA database
- Configure when log backups are created
- Recover SAP HANA database

Note:
This backup and recovery documentation covers only SAP HANA database. It does not describe how to perform a backup and recovery of all the components that can be part of a SAP HANA installation.

Note:
This documentation does not replace the documentation that you are recommended to use for your specific production operations.
13.2 Prerequisites for Backup and Recovery

To perform backup operations, the following requirements must be met:

- Backups can only be executed by database users that have the following system privileges:

<table>
<thead>
<tr>
<th>Authorization for Backup</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACKUP ADMIN</td>
</tr>
<tr>
<td>Authorization to perform backups</td>
</tr>
<tr>
<td>CATALOG READ</td>
</tr>
<tr>
<td>Authorization to collect the</td>
</tr>
<tr>
<td>information needed by the backup</td>
</tr>
<tr>
<td>wizard</td>
</tr>
</tbody>
</table>

- Other privileges are described in the following documentation:
  
  *SAP HANA Security Guide*
  
  *SAP HANA SQL Reference*

- The recovery process is executed as operating system user `<sid>adm`

  **Note:**
  
  During installation, the database user `SYSTEM` is created. The `SYSTEM` user has all database privileges, but is not allowed to see the data owned by other users. It is recommended that the `SYSTEM` user NOT be used for normal database operation. Instead, it is recommended that you create dedicated database users with only the privileges necessary for backup and recovery.

Related Links

*SAP HANA Security Guide*

*SAP HANA SQL Reference*

13.3 Points to Note About Backup and Recovery

Below is an overview of important points to consider when planning your backup and recovery strategy with SAP HANA database. You can find more information on the individual points in the subsequent sections of this documentation.

**Points to Note: Backup**

- Data and logs can only be backed up when SAP HANA database is online (when all configured services are running).
- During data and log backups, the database is available as usual.
- Data and log backups are performed independently of each other.
- In a data backup, only the database payload is backed up.
- Until an initial data backup has been completed, the logs are written in overwrite mode and no log backups are made.
• Backup and recovery always apply to the whole database.
  It is not possible to back up and recover individual database objects.
• SAP HANA database can be backed up to the file system or using third-party backup tools.

Points to Note: File-Based Backup
• The configuration path for data and log backups must be valid throughout the whole system, not only for specific hosts.
• Shared storage must be used.
  This is to ensure that the nameserver process can access the backup files at the time of recovery.

Points to Note: Recovery
• To recover a SAP HANA database, the database needs to be shut down.
  For this reason, during recovery, the database cannot be accessed by end users or applications.
• A SAP HANA database cannot be recovered to a SAP HANA database with an lower software version.
  The SAP HANA database software version used for the recovery must always be the same version or higher than the SAP HANA database used to create the backup.
• To recover the database, you need at least one data backup.
• At the beginning of a recovery, all the data and log backups to be used must be either accessible in the file system or available through the third-party backup tool.
• Once started, a database recovery cannot be interrupted.
• If a recovery fails, the complete recovery must be repeated.

Backup and Recovery Capabilities
Below is an overview of the backup and recovery capabilities for SAP HANA database:

<table>
<thead>
<tr>
<th>Capability</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Backup</td>
<td>YES</td>
</tr>
<tr>
<td></td>
<td>More information: About Data Area Backup</td>
</tr>
<tr>
<td>Log Backup</td>
<td>YES</td>
</tr>
<tr>
<td></td>
<td>More information: About Log Area Backup</td>
</tr>
<tr>
<td>Backup of files with customer-specific configuration settings</td>
<td>NO</td>
</tr>
<tr>
<td></td>
<td>More information: Backing Up the SAP HANA Configuration Files</td>
</tr>
</tbody>
</table>
### 13.4 About SAP HANA Database Backup

A SAP HANA database backup consists of the following:

- **Backup of the data area**
  
  This is done from persistent storage to the backup destinations. Backups are done either manually or are scheduled.

- **Backup of the log area**
  
  By default, logs are backed up automatically. You can configure whether and how the logs are backed up.

  **Note:** Data and log backups are performed independently of each other.

To ensure that it is possible to recover the database, it is recommended that you perform a data backup in the following situations:

- After the initial load
- At regular intervals

  **Note:**

  You can use less recent data backups for a recovery, provided that the subsequent log backups are available. If more log backups have to be replayed, the recovery takes longer to complete. For this reason, to recover the database, it is recommended that you use the most recent data backup and subsequent log backups. The more frequently a database is backed up, the faster the recovery will be.

  When you recover the database to a specific data backup, you can no longer apply backups older than the backup used for the recovery.

- Before the database software is upgraded to a new version
  
  If a software upgrade fails, it is possible to use this backup to recover the database to its state before the upgrade.

- If a new host is added to the landscape

  **Note:**
After a new host is added to the landscape, the redo log history is interrupted because the landscape has changed. For this reason, it is no longer possible to use the redo log to recover the database. After a new host has been added, it will be possible to use the existing database and log backups to recover the database only to the point before which the host was added.

If you add a new host to the system, you should also stop production operation until the host has been added and a new complete data backup has been made.

- After any situation that causes log writing to be interrupted
  
  For example, immediately after log writing is enabled or after a data import.

### 13.4.1 About Data Area Backup

When the data area is backed up, the payload data is backed up for all the SAP HANA services running. If SAP HANA is running on multiple hosts, the data backup includes all the service-specific backup parts for all the hosts.

#### Default Destination for the Data Backups

By default, data backups are written to the following destination: `$DIR_INSTANCE/backup/data`.

You can specify a different destination when you perform the backup. Alternatively, you can change the default backup destination in the `Backup` section in SAP HANA studio.

To change the default destination for data backups:

1. In SAP HANA studio, go to the `Backup` section.
2. Go to the `Configuration` tab and choose `Data Backup Settings`.
3. Specify the new default destination in the `Destination` field.
4. Save.

**Note:** Changes to the default backup destination take effect immediately.

For improved data safety, it is recommended that you specify a path to an external backup destination. The backup destination should never be on the same file system as the data or log areas.

**Note:** The default backup destination can only be changed for file-based backups. Backups made using third-party tools always use the destination `/usr/sap/<SID>/SYS/global/hdb/backint`. For this reason, it is not possible to change the backup destination for third-party tools. It is recommended to assign individual backup names to be able to easily identify the backups.

**Note:** Different data backups can be written to different destinations, but all the parts of one particular data backup are written to the same destination. For file-based backups, it is recommended that you create the destination directory structures before the backup is started.

#### Backup Names

Each backup name comprises the following elements: `<<path><prefix>>`
Table 16: Elements of Backup Names

<table>
<thead>
<tr>
<th>Name Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;&lt;path&gt;&gt;</td>
<td>Optional. If no complete path is specified, the default backup destination is prepended to the backup name. For file-based backups, the backups are written to a destination in the file system. For backups made using third-party backup tools, a pipe is created in the file system. The pipe is always created in the directory /usr/sap/&lt;SID&gt;/SYS/global/hdb/backint. The backup tool decides where the backups are actually written.</td>
</tr>
<tr>
<td>&lt;&lt;prefix&gt;&gt;</td>
<td>Optional. You can use the prefix proposed by the system or you can specify a different prefix for the backup name.</td>
</tr>
</tbody>
</table>

**Note:** For file-based backups, it is recommended that you use unique name prefixes for each data backup. For example, a unique timestamp. If you use the same prefixes, it is recommended that you copy a data backup to a new destination as soon as the backup is created. Otherwise, an existing data backup with the same name will be overwritten by the next data backup. Alternatively, when you start the next backup, specify a different backup prefix or path. For backups made using third-party backup tools, data backups will not be overwritten. The Backint for SAP HANA interface is able to identify multiple versions of backups with the same name.

**Note:** The system adds a unique suffix to each backup name. As this is done for each service that is included in the backup, you only need to specify the name (<<path><prefix>>) for all the backups on the different hosts. The next time a service is backed up, the system assigns the same suffix to the backup for that service.
Note: Once backups have been created, it is strongly recommended that you do not change their names. When they are created, backup names are stored in the backup catalog. In a recovery scenario, specific backup components are located using the names stored in the backup catalog. If the name of a backup was changed after it was created and recorded in the backup catalog, it will not be possible to locate it using the backup catalog.

You must not change the prefix. For file-based backups, you can change the path. The backup files are then written to a different destination in the file system. In a recovery situation, you need to specify the appropriate path.

Names for Parts of a Data Backup

During backup, each service backs up its data to the specified backup destination. Below is an example of a set of backups from one data backup created with SAP HANA studio. The backups can have different names.

```plaintext
<backup/data/COMPLETE_DATA_BACKUP_databackup_0_1> (name server topology)
<backup/data/COMPLETE_DATA_BACKUP_databackup_1_1> (name server)
<backup/data/COMPLETE_DATA_BACKUP_databackup_2_1> (for example, index server)
```

In the above example, the `<path>` is `</backup/data/`, the `<prefix>` is `<COMPLETE_DATA_BACKUP>`, `<databackup_0_1>` is the suffix, which is automatically added by the system.

Estimating the Space Needed for Backups

The SAP HANA Administrator has to ensure that sufficient free space is available for the backups. If insufficient space is available for a backup, the backup will fail. For this reason, it is necessary to estimate the amount of space that will be needed in the backup destination or on the third-party backup server.

Note: The actual size of a data backup can be larger or smaller than the estimated size. For example, if data is changed in the database after the size has been estimated and before the backup is performed, the backup may be different from the estimated size.

To estimate the size of the next complete data backup, you can use either of the following commands:

- select sum(allocated_page_size) from M_CONVERTER_STATISTICS
  
The result is a single value that gives the sum of the sizes of all services in bytes.

  The size may differ between `SELECT` statement and `DATA Back up` execution. For this reason, it is advisable to include a reserve of free space.

- select volume_id, sum(allocated_page_size) from M_CONVERTER_STATISTICS group by volume_id

  The size of the next complete data backup will be larger or smaller than the estimated size.
This displays a list of the volumes (index server, name server, statistics server), with the size of each volume in bytes.

Both commands use the system table M_CONVERTER_STATISTICS, which contains information about the blocks used.

13.4.2 About Log Area Backup

The system can perform regular automatic backups of the redo logs. During a log backup, the payload of the log segments is copied from the log area to service-specific log backups or to a third-party backup server.

A log segment is backed up in the following situations:

- The log segment is full.
- The log segment is closed after exceeding the configured time threshold.
- The database is started.

Caution:

Log segments are automatically re-used for new log entries. Log segments can only be reused after they have been backed up.

Default Destination for the Log Backups

By default, file-based log backups are written to `$DIR_INSTANCE/backup/log`.

To change the default destination for file-based log backups:

1. In SAP HANA studio, go to the Backup section.
2. Go to the Configuration tab and choose Log Backup Settings.
3. Specify the new default destination in the Destination field.
4. Save.

Note: Changes to the default log backup destination take effect immediately.

Note: To improve data safety, the log backup destination should never be on the same file system as the data area or the log area.

The default backup destination can only be changed for file-based backups. Backups made using third-party tools always use the destination `/usr/sap/<SID>/SYS/global/hdb/backint`. For this reason, it is not possible to change the backup destination for third-party tools.

Log Backup Options

You can configure the log backup behavior using the following parameters in the `global.ini` configuration file.

Note: To configure log backup behavior, you need the privilege `CATALOG READ`.

To change these parameters in the global.ini configuration file:

1. In SAP HANA studio, go to Administration ➔ Configuration ➔
2. Open `global.ini` > `persistence`.
3. Double-click a parameter to change it.
4. When you have made your changes, choose Save.

**Note:**
When you change the log mode, you must restart the database system to activate the changes. It is also recommended that you create a full data backup of the database.

### Automatic Log Backup

Automatic log backup can be enabled or disabled using parameter `enable_auto_log_backup`.

Default: `enable_auto_log_backup = yes`

**Note:** During normal system operation (log mode normal), it is recommended that automatic log backup be kept activated. If automatic log backup is disabled, the log area grows until the file system is full. If the file system is full, the database will freeze.

More information: *Log Modes*

If the database freezes:

1. Back up the log area.
2. Use the SQL command `ALTER SYSTEM RECLAIM LOG` to physically delete log segments that are no longer needed. When automatic log backup is enabled, it is necessary only in exceptional situations to use the `ALTER SYSTEM RECLAIM LOG` command. In log mode normal, a database freeze is probably the only situation, in which it is necessary to reclaim the log in this way. As it can take some time to reformat the blocks in the log area, using `ALTER SYSTEM RECLAIM LOG` in log_mode normal would normally be counterproductive.

### Log Modes

**Note:**
When you change the log mode, you must restart the database system to activate the changes. It is also recommended that you create a full data backup of the database.

The log_mode parameter can have the following values:

- `log_mode = normal`

  Normal mode (default)

  Log segments are automatically backed up if parameter `enable_auto_log_backup` is enabled.

  `log_mode = normal` is recommended to provide support for point-in-time recovery.

  Automatic log backups can prevent log full situations from arising.
After the system has backed up the full log segment, the system can reuse the space that the full log segment occupied in the log area to overwrite it with new log entries. If the log area does become full and no more log segments can be created on disk, a log full situation arises and the database freezes. When the log area is full, no more log entries can be written until a log backup has been done.

**Caution:**
Do not delete log segments on operating system level, as this will make the log area unusable. As a result, the database may stop working immediately.

- **log_mode = overwrite**

  Overwrite mode

  Log segments are freed by savepoints and no log backup is performed. For example, this can be useful for test installations that do not need to be backed up or recovered.

  **Caution:**
  
  log_mode = overwrite is not recommended for production systems.

  With log_mode = overwrite, no point-in-time recovery is possible. For recovery, only data backups are used; the logs are not used. Only the following recovery option can be selected: Recover the database to a specific data backup.

- **log_mode = legacy**

  In legacy mode, no log backup is performed.

  Log segments are retained until a full data backup is performed. This is to allow recovery from the most recent full backup and the log in the log area.

  This was the default setting for SAP HANA SPS 02.

  **Note:**
  
  log_mode = legacy is still supported, but not recommended. After an upgrade from SAP HANA SPS 02 to SPS 03 or higher, it is recommended that you use log_mode = normal.

**Switching From Log Mode Legacy to Normal**

When you switch log mode from legacy to normal, check the following settings:

- Automatic log backup is enabled.
- The log backup interval is greater than 0.

  For example, a value of 15 minutes ensures that if you need to recover the database and if the log area is lost, you will lose no more than 15 minutes of data.

To change the log backup settings:
1. In SAP HANA studio, go to the Backup section.
2. Go to the Configuration tab and choose Log Backup Settings.
3. Make the changes.
4. Save.

**Note:**
When you switch to log mode normal, log backups are created on a continuous basis. Over time, this requires a steadily increasing amount of disk space. If the situation is not managed properly, the disk space could become full at some stage. For this reason, we recommend that you change the log and data backup destinations to a separate storage medium with sufficient disk space. Alternatively, you could consider using a third-party backup solution. In addition to helping to prevent space issues, using a separate storage medium also improves safety in case of hardware failures, and performance.

More information: SAP Note 1645183 - Changing log mode for SAP HANA DB after SPS2 to SPS3 upgrade

**Log Backup Timeout**

You can specify a timeout for log backups. The timeout determines the intervals at which the logs are backed up.

**Note:** If the log segments become full before the timeout, the logs will be backed up anyway.

Specifying an appropriate timeout for log backups enables you to recover a SAP HANA database with minimum data loss. For example, if you need to recover the database in a situation where the log area is unusable, and only the data and log backups are available.

Recommended: Specify a timeout. For example, 15 minutes.

(If you specify a timeout 0, the log backups are only made when a log segment is full and when services are restarted.)

**Note:**
The backup timeout only takes effect if automatic log backups are enabled.

To change the log backup timeout:
1. In SAP HANA studio, go to the Backup section.
2. Go to Log Backup Settings.
3. Specify the desired log backup timeout interval.
4. Save.

**More information**

ALTER SYSTEM RECLAIM LOG in the SAP HANA SQL Reference

Location of the Log Backup Files
13.5 Working with Third-Party Backup Tools

In addition to backing up the SAP HANA database to files in the file system, you can back up and recover the SAP HANA database using a third-party tool that supports the Backint for SAP HANA interface. Third-party backup tools communicate with the SAP HANA database through the Backint for SAP HANA interface. Backint for SAP HANA uses pipes to back up the content of the database and performs all the actions needed to write the backup data to external storage.

As SAP HANA database is a distributed database, each active host in the database may have one or more volumes to be backed up. When Backint for SAP HANA is used to back up a database, several Backint communication processes are started, one for each volume.

Backint for SAP HANA is supported for the SAP HANA server (Linux only).

For more information about installing and configuring a third-party backup tool, consult the documentation provided by the tool vendor.

Note: Backup tools using Backint for SAP HANA must be certified by SAP. For more information, see SAP Note: 1730932 (Using backup tools with Backint for HANA)

13.5.1 Configuring Third-Party Backup Tools

You can specify whether data and log backups are created to the file system or using third-party backup tools. The default configuration for a third-party backup tool is defined when the tool is installed. After a backup tool has been installed, you can back up and recover the SAP HANA database without making any further changes. However, you have the option to make certain configuration changes for the backup tool.

1. In SAP HANA studio, open the Backup section for the database. If a backup is currently running, its progress is displayed.
   Performing a Data Backup Using SAP HANA Studio

2. Go to the Configuration tab. The current settings are displayed for file-based data backup and log backup. For file-based data backup, you can change the default destination. For log backup, you can specify the destination type, the destination, and the backup interval. For third-party backup tools (Backint), the backup destination is always /usr/sap/<SID>/SYS/global/hdb/backint for both data backup and log backups. For third-party backup tools, you can only specify the log backup interval.
   A third-party backup tool only uses the backup destination to create pipes to determine where the backups are written to. The pipes occupy only a negligible amount of space in the file system. As SAP HANA database queries information from the third-party backup tool during recovery, the backup destinations cannot be changed.

3. To configure Backint, open the Backint Settings section. If a Backint agent is currently active, it is displayed.
   You cannot change the Backint agent using SAP HANA studio. If required by the tool vendor, you can specify Backint parameter files for data backup and for log backups. The content and syntax of the parameter files is tool-specific and defined by the tool vendor. For more information, see the documentation for the third-party backup tool.

   If you disable Backint, you need to check that the destination used for file-based backups is correct.
13.6 Backing Up the SAP HANA Configuration Files

The configuration files contain the SAP HANA database configuration settings.

The configuration files are not backed up as part of the database backup. Configuration files that contain customer-specific changes can be backed up manually. In a recovery situation, this can be helpful to more easily identify and restore the customer-specific changes.

The configuration files are not essential to perform a recovery. If you want to use a customer-specific configuration, you need to reconfigure the recovered system using SAP HANA studio. To display the configuration values, go to the Configuration tab in SAP HANA studio. The configuration files all have the extension `.ini`.

Binary Configuration File

In addition to the configuration files, all customer-specific changes are also saved in one separate (binary) configuration file. This file is created when SAP HANA is installed and is stored in the same directory as the configuration files.

The binary configuration file is versioned. When the file is changed, a new version is created and the previous version is renamed sequentially. All the file versions are stored in the same directory.

If you want to back up customer-specific configuration changes, you should back up all the versions of the binary configuration file manually together with the other configuration files.

In a recovery scenario, if you wish to restore customer-specific settings, you can use both the configuration files (.ini files) and the binary configuration file.

To restore customer-specific configuration settings from the binary file, use the command line tool hdbparam. If you do not want to restore the most recent version of the binary file, use hdbparam to check the individual parameter values and decide which version of the binary file to restore.

Note: The binary configuration file does not contain the information from sapprofile.ini.

More Information

Copying a Database Using Backup and Recovery

13.6.1 Location of the Configuration Files

The configuration files (.ini files) are located by default in the following directories:

<table>
<thead>
<tr>
<th>Path</th>
<th>Configuration Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>For global configuration settings:</td>
<td>custom/config global.ini</td>
</tr>
<tr>
<td>$DIR_INSTANCE/../SYS/global/hdb/</td>
<td>indexserver.ini</td>
</tr>
</tbody>
</table>

Table 17: Locations of the SAP HANA Configuration Files
13.6.2 Configuration Files Created During Installation

During installation of SAP HANA database, the following customer-specific configuration files are created:

Table 18: Content of Customer-specific SAP HANA Configuration Files

<table>
<thead>
<tr>
<th>Configuration File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sapprofile.ini</td>
<td>Contains system identification information, such as the system name (SID) or the instance number. After installation, sapprofile.ini is not changed again.</td>
</tr>
<tr>
<td>daemon.ini</td>
<td>Contains information about which database services to start.</td>
</tr>
<tr>
<td>nameserver.ini</td>
<td>The nameserver.ini file contains global information for each installation. The landscape section contains the system-specific landscape ID and assignments of hosts to roles MASTER, WORKER, and STANDBY. If the system landscape is changed using hdbaddhost, the landscape section of nameserver.ini is also changed.</td>
</tr>
</tbody>
</table>

### Note:
- Configuration files (.ini files) are only created in the above directories if customer-specific changes are made to them after installation. If no customer-specific changes have been made, these directories may be empty.

13.7 About the Backup Catalog

The backup catalog provides information about the execution and history of backups. The backup catalog enables the system to do the following:
- Determine whether a recovery is possible
- Determine which data and log backup to use to recover the database
- Determine which backups are no longer needed

The backup catalog includes the following information:

- Backups performed for a database
- The start and finish times of the backups
- Whether a backup is still running
- Whether a backup was successful or not
- Volumes that were backed up
- Log backups and which part of the log they contain
- Backup destinations and their sizes
- Whether the redo log history was interrupted
- The destination type (File or Backint)
- An external backup ID (if a third-party backup tool is used)

**Note:** The backup catalog does not show the progress of a backup. The progress of a backup is recorded in `backup.log`.

The backup catalog is stored in the SAP HANA database and is part of the redo log backup. This allows access to the backup catalog during recovery. Even in situations such as when `log_mode = overwrite` is set, where logs are not written, the backup catalog is still backed up. The backup catalog is backed up and versioned after every completed backup operation. If the backup catalog is saved using a third-party backup tool, the versioning of the backup catalog is handled by the backup tool.

**Related Links**

*Diagnostics Files for Backup and Recovery* [page 175]

### 13.7.1 Monitoring Views for the Backup Catalog

The monitoring views `M_BACKUP_CATALOG` and `M_BACKUP_CATALOG_FILES` can be used to display information about the backup catalog.

**Note:** To use the `M_BACKUP_CATALOG` and `M_BACKUP_CATALOG_FILES` views, the system privilege `CATALOG READ` is needed.

**Table 19: Monitoring Views for the Backup Catalog**

<table>
<thead>
<tr>
<th>Monitoring View</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>M_BACKUP_CATALOG</td>
<td>Provides an overview of information about backup activities. Each row in the view provides information for a separate catalog entry that is identified by a backup ID. This information includes the type (complete data backup, log backup and log missing), and start and completion times.</td>
</tr>
</tbody>
</table>

**Note:**
The backup ID is used to reference the parts of a backup in the M_BACKUP_CATALOG_FILES view.

M_BACKUP_CATALOG_FILES provides information about the backups created, and the backup destinations that are used by data and log backups.

Each row in the view has a corresponding entry in M_BACKUP_CATALOG. Each row is identified by a backup ID.

M_BACKUP_CATALOG_FILES provides additional information about each database service that was involved in a backup. For example, with a data backup, each database service is listed with its specific backup information such as destination path and redo log position.

13.7.2 Using the Backup Catalog to Identify Log Backups That can be Deleted

Log backups that are older than the last available data backup can be deleted. To find out which log backups can be deleted, you can use the following system views:

- M_BACKUP_CATALOG
- M_BACKUP_CATALOG_FILES

Note:

Deleting log backups does not change the backup catalog. If log backups are deleted, the corresponding entries remain in the backup catalog.

Log files should never be deleted manually.

Each service-specific data backup file in the complete data backup has its own redo log position. The redo log position identifies the log backup that belongs to the service-specific data backup file. The log position for each service-specific data backup file is contained in M_BACKUP_CATALOG_FILES.REDO_LOG_POSITION.

Additionally, each service-specific log backup contains the range of log entries that are in the backup:

- The oldest log entry in a log backup is specified by M_BACKUP_CATALOG_FILES.FIRST_REDO_LOG_POSITION
- The most recent entry in a log backup is specified by M_BACKUP_CATALOG_FILES.LAST_REDO_LOG_POSITION

If you want to delete old log backups, you have to decide which complete data backup is the oldest that you potentially want to recover. You can use the redo log position to find all the older service-specific log backups. Using M_BACKUP_CATALOG_FILES, search for all service specific log backups, for which the LAST_REDO_LOG_POSITION is smaller than the REDO_LOG_POSITION of the corresponding data backup.
This is an example statement to find older service-specific log backups:

```sql
select destination_path from m_backup_catalog_files o where o.last_redo_log_position is not null and exists(select 1 from m_backup_catalog_files i where i.backup_id = <backup_id> and i.source_id = o.source_id and i.redo_log_position > o.last_redo_log_position) order by destination_path
```

Note:
To find the `<backup_id>` of the requested complete data backup, query the monitoring view M_BACKUP_CATALOG.

### 13.8 Diagnostics Files for Backup and Recovery

The `backup.log` and `backint.log` files record information about backups and recoveries. This information can be used to diagnose errors.

**Note:**
As more information is written to `backup.log` and `backint.log`, they grow, but their increased size does not impact database performance. If `backup.log` or `backint.log` do become too big, the Administrator can delete them as required. However, it is not recommended to delete the files regularly or automatically.

**backup.log**

The `backup.log` records information about the data backups, the log backups, and the backup catalog.

To open `backup.log`:

1. From the SAP HANA studio, choose *Diagnosis Files*.
2. Search for `backup.log` in the *Filter* field.
3. Open the file.

**backint.log**

`backint.log` contains information about the activities of the Backint agent. Backint agent is part of a third-party backup tool. `backint.log` records all the parameters used to call the Backint agent and the values returned. Each time the Backint agent is called, the command parameters and the return code are appended to `backint.log`.

`backint.log` consists of the content of the following:

- Backint input file
  - This file is created when the Backint agent is started.
- Backint output file
The Backint agent writes its output to this file. The contents of the command file and the output file are copied to backint.log.

13.9 Backing Up the SAP HANA Database

The sections that follow describe how to perform a data backup using different tools:

- SAP HANA studio
- SQL commands
- Batch mode

**Note:**
You can also schedule periodic backups using DBA Cockpit. To work with DBA Cockpit, you need a compatible ABAP system.

**More information**

DBA Cockpit documentation on the [SAP Help Portal](https://help.sap.com/).  

13.9.1 Performing a Data Backup Using SAP HANA Studio

**Note:**
A data backup performed with SAP HANA studio saves only the payload of the database. The database configuration files (*.ini files) are not backed up. Configuration files that contain customer-specific changes can be backed up manually in order to more easily identify and restore customer-specific changes in a recovery situation.

**More information:** [About the SAP HANA Configuration Files](https://help.sap.com/).

To create a data backup:

1. Open SAP HANA studio.
2. In the **Navigator** view, select the database that you want to back up.
3. From the context menu, choose **Back Up...**
4. Select a destination type:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>Backs up the database to files in the file system. Each SAP HANA service writes backup data to the specified destination in the file system.</td>
</tr>
<tr>
<td>Backint</td>
<td>Backs up the database using a third-party backup tool. Each SAP HANA service starts the Backint agent and sends the backup data to the third-party backup tool.</td>
</tr>
</tbody>
</table>

5. Specify the backup destination and the backup prefix.

The default destination is the path specified in the **Backup** section. The backup destination can be the same, regardless of whether you select **File** or **Backint**. The third-party backup tool can distinguish between multiple backups with the same name. You do not need to use a different name for each backup. For file-based
backups, you need to ensure that there is sufficient space at the specified backup destination. For backups with third-party backup tools, you can only change the backup prefix. For third-party backup tools, the destination is always /usr/sap/<SID>/SYS/global/hdb/backint.

6. Choose Next. A summary of the backup settings is displayed.

7. When all the settings are correct, choose Finish.

The backup then starts.

The progress of the backup is shown for all the services (for example, the statistics server, name server, and index servers).

When all the volumes have been backed up, a confirmation message is displayed.

### 13.9.2 Performing a Data Backup Using SQL Commands

You can enter SQL commands either by using the SQL console in SAP HANA studio, or by using the command line program `hdbsql`.

**Note:**

Backups using SQL commands are only recommended for batch mode.

More information: *Performing a Data Backup in Batch Mode*

You can start the backup using the following SQL commands:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>For file-based backup:</td>
<td><code>BACKUP DATA USING FILE ('&lt;path&gt;&lt;prefix&gt;')</code></td>
</tr>
<tr>
<td>For third-party backup tools:</td>
<td><code>BACKUP DATA USING BACKINT ('/usr/sap/&lt;SID&gt;/SYS/global/hdb/backint/&lt;prefix&gt;')</code></td>
</tr>
</tbody>
</table>

**Note:** For third-party backup tools, you can optionally omit the path specification.

If no absolute path is specified, the default backup destination is prepended to the location.

```
BACKUP DATA USING FILE ('/backup/data/MONDAY/COMPLETE_DATA_BACKUP')
```

This would create the following files in the directory `/backup/data/MONDAY`:

- COMPLETE_DATA_BACKUP_databackup_0_1 (name server topology)
- COMPLETE_DATA_BACKUP_databackup_1_1 (name server)
- COMPLETE_DATA_BACKUP_databackup_2_1 (for example, index server)

... 

**Note:** If the directory does not exist, you need to create it before you start the backup.

More information: SAP Note [1651055 - Scheduling SAP HANA Database Backups in Linux](#)
13.9.3 Performing a Data Backup in Batch Mode

To perform data backups in batch mode, it is recommended that you use the command line tool hdbsql on operating system level. hdbsql enables you to trigger backups through crontab.

**Note:**

For batch processing, hdbuserstore is required. hdbuserstore is installed as part of the client package. It is recommended that you set up a batch user with backup privileges (BACKUP ADMIN).

**Caution:**

It is not recommended that you start the backup command using hdbsql on operating system level without hdbuserstore. This is because the password used for hdbsql connect will be visible in the process list.

Performing a Data Backup Using Cron

1. Install the client software. The client software enables access to the hdbuserstore. Use the following command: hdbinst -a client (default location: /usr/sap/hdbclient)
2. Create a user key. Use the following command: /usr/sap/hdbclient/hdbuserstore set <KEY> <host>:3<instance id>15 <user> <password>.
   
   Example: /usr/sap/hdbclient/hdbuserstore set BACKUP vebwtests1:30015 user password
3. In crontab, call the following at the desired time: /usr/sap/hdbclient/hdbsql -U<KEY> "BACKUP DATA USING FILE ('<path><prefix>')"
   
   Example: /usr/sap/hdbclient/hdbsql -U BACKUP "BACKUP DATA USING FILE ('MONDAY')"

A data backup is then created in the default destination. In the above example, the prefix of all service-related backup files is 'MONDAY'.

13.9.4 Canceling a Running Data Backup

To cancel a data backup that is still running, a user needs the system privileges CATALOG READ and BACKUP ADMIN.

To cancel a running data backup, you can use either SAP HANA studio or an SQL command.

**Canceling a Running Data Backup Using SAP HANA Studio**

You can use SAP HANA studio to cancel running data backups that you started using the backup wizard.

1. In SAP HANA studio, go to the Backup section.
2. Go to the Overview tab and open Status of Currently Running Backups.
3. Choose Cancel Backup.

The backup is then canceled and you are notified of this.

**Canceling Running Data Backups Using SQL Commands**

To cancel any data backup:
1. Find the backup ID. When a data backup is started, the system assigns a unique ID to the data backup.
   a) To find the backup ID of the running data backup, use the monitoring view M_BACKUP_CATALOG, which
      provides an overview of information about backup and recovery activities.
   b) To find the backup ID of the running data backup, use the following SQL command:
      \[\text{select BACKUP ID from "M_BACKUP_CATALOG" where entry_type_name = 'complete data backup' and}
      \text{state_name = 'running' order by sys_start_time desc;}\]
   c) You can now use the backup ID to cancel the running data backup.

2. Cancel the data backup. To cancel the running data backup use the following SQL command:
   \[\text{backup cancel <backup_id>.}\]

3. Check that the data backup was canceled. The data backup is canceled asynchronously.
   a) To check that the running data backup was canceled, check the monitoring view M_BACKUP_CATALOG.
      This view shows the state of the canceled data backup.
   b) If its state is "canceled", this confirms that the backup was canceled successfully. When the backup state
      is "canceled", you can start a new data backup.

   **Note:** "cancel pending"

   If you cancel a running data backup that is waiting for other resources, the cancelation is postponed until
   the cause of the situation has been resolved. Until that time, the backup is flagged as "cancel pending".

   To check the state of the data backup, use the following command:

   \[\text{select state_name from "SYS"."M_BACKUP_CATALOG" where backup_id = <backup_id>}\]

### 13.10 About SAP HANA Database Recovery

It may be necessary to recover the SAP HANA database in the following situations:

- A disk in the data area is unusable.
- A disk in the log area is unusable.
- As a consequence of a logical error, the database needs to be reset to its state at a particular point in time.
- You want to create a copy of the database.

#### 13.10.1 Recovery Scenarios

The steps to recover the database depend on the recovery scenario and the reason for the recovery. This section
provides an overview of recovery scenarios.

**Data Area is Unusable**

If the data area is unusable, and all the data changes after the last complete data backup are still available in the
log backups and log area, the data from committed transactions that was in memory at the time of failure can be
recovered.

No committed data is lost.
For recovery, the data backups, the log backups, and the log area are used. When the data backup has been successfully recovered, the log entries from the log backups and the log area are automatically replayed.

It is also possible to recover the database using an older data backup and log backups. All relevant log backups made after the data backup are needed for the recovery.

More information: SAP Note 1705945 (Determining the files needed for a recovery)

Note:
In the Recovery Wizard dialog, ensure that the paths to the data and log backup files are correct.

More information:
Default Destination of the Data Backup Files
Default Destination of the Log Backup Files

Used for Recovery
- Data backups
- Log backups
- Log area

Steps for Recovery
Perform a database recovery to the most recent state.

Log Area is Unusable
If the log area is unusable, it is only possible to replay the log backups. As a consequence, any changes that were made after the most recent log backup will be lost. In addition, all the transactions that were open during the log backup will be rolled back.

It is still possible to recover the database to a point in time within the existing log backups.

More information: Log Backup Options

For recovery, the data backups and the log backups are used. When the data backup has been successfully recovered, the log entries from the log backups are automatically replayed. In the Recovery Wizard, you must specify the option Initialize log area to prevent the recovery of entries from the unusable log area. This option initializes the log area, and the old content of log area is lost.

Used for Recovery
- Data backups
- Log backups

Steps for Recovery
1. Perform a database recovery to the most recent state.
2. Specify the option Initialize log area.
Logical Error – Point in Time Recovery

To reset the database to a particular point in time, you need a data backup from before the point in time to recover to, the subsequent log backups, and the log area.

All changes made after the recovery time will be lost. If you need to perform this recovery, consider recovering the database to a different system.

Used for Recovery

- Data backups
- Log backups
- Log area

Steps for Recovery

Perform a database recovery to a point-in-time before the logical error.

Note:

You need to specify a point in time to recover the database to. If you specify a point in time in the future, the effect is the same as recovering the database to the most recent state.

13.11 Recovering the SAP HANA Database

To recover a SAP HANA database, the following requirements must be met:

- The SAP HANA database software must be installed, so that an empty database exists.

  Note:

  If you want to restore customer-specific configuration settings, it is recommended that you do this before you recover the database and the log backups. In a recovery situation, you can use SAP HANA studio to restore customer-specific configuration settings.

- Ensure that the target system and the source system have identical configurations.

  The number of hosts and the number and types of services (for example, index server) on each host must be identical for both system landscapes.

  Note:

  For information about copying a SAP HANA database from multiple hosts to a single host:

  Note 1749467 - Copying SAP HANA From a Multiple- to a Single-Host System

- At least one data backup must be available before the recovery is started.
- To recover the database to a particular point in time, a data backup and all the log backups up to the point in time for recovery are needed.
To shut down the database, you need to know the password of the operating system user `<sid>adm`.

The following constraints apply:

- Recovery to a lower system release is not possible.
- Once it is running, a recovery cannot be canceled.
- If an error occurs during a recovery, the complete recovery must be repeated.

The recovery process is started using SAP HANA studio.

---

### Note:

To recover the SAP HANA database, the database needs to be shut down. For this reason, during recovery, the database cannot be accessed by end users or applications.

---

To recover a SAP HANA database:

1. From SAP HANA studio, open the context menu for the database to be recovered.
2. Choose **Recovery...**
   
   You are requested to confirm that the system can be shut down for the recovery.
3. Confirm and enter the `<SID>adm` user and password.
4. Choose **OK**. The database is shut down.
5. Specify the recovery type. The following recovery types are available:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recover the database to its most recent state</td>
<td>This option recovers the database to as close as possible to the current time.</td>
</tr>
<tr>
<td></td>
<td>This recovery option uses the following data:</td>
</tr>
<tr>
<td></td>
<td>○ The most recent data backup</td>
</tr>
<tr>
<td></td>
<td>○ Log backups made since the most recent data backup</td>
</tr>
<tr>
<td></td>
<td>○ Log area</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Recover the database to the following point in time</th>
<th>This recovery option uses the following data:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>○ The last data backup available before the specified point in time</td>
</tr>
<tr>
<td></td>
<td>○ Log backups made since the data backup to be used</td>
</tr>
<tr>
<td></td>
<td>○ Log area</td>
</tr>
</tbody>
</table>

### Note:

If you specify a point in time in the future, the effect will be the same as recovering the database to the most recent state.

<table>
<thead>
<tr>
<th>Recover the database to a specific data backup</th>
<th>This recovery option uses the following data:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>○ The specified data backup</td>
</tr>
</tbody>
</table>

**Caution:** The SAP HANA database will be initialized with the specified data backup. This backup becomes the first backup of a new database life cycle. Older backups are not compatible with the log written after the recovery.

**Caution:** Log entries are not replayed, neither from the log backups nor from the log area. All log entries that still exist in the log area are deleted. All the changes that were made after this backup will be lost.
Option Description

1. **Note:** You need to know the destination type (File or Backint), the location, and the prefix of the data backup.

2. **Note:** This recovery type is an advanced option that can be used in exceptional cases if a previous recovery failed.

   This recovery option uses the following data:
   1. The most recent data backup available before the specified log position
   2. Log backups made since the data backup to be used
   3. Log area

6. Choose **Next**.

   The SAP HANA database uses the backup catalog to identify the location of the backups. You do not need to specify whether **File** or **Backint** was used for the backup.

7. Depending on the recovery type, provide the necessary information:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Recover the database to its most recent state</strong> OR <strong>Recover the database to the following point in time</strong> :</td>
<td>Specify the location of the log backup files if they are not in the default location.</td>
</tr>
<tr>
<td><strong>Recover the database to a specific data backup</strong></td>
<td>Select the destination type (File or Backint) For file-based backups, specify the location and the prefix of the data backup files.</td>
</tr>
</tbody>
</table>

8. Choose **Next**.

9. Select the option **Install new license** and specify the license key file for the recovered database.

   You can specify the option **Initialize log area**. With this option, no log entries from the log area are replayed, and the log area is initialized. The content of the log area is lost. The log entries from the log backups are replayed if they are needed.

   In the following situations, you must specify the option **Initialize log area**:
   1. If the log area is unusable
   2. If you are recovering the database to a different system

10. Choose **Next**.

   A summary of the selected recovery options is displayed. To make changes to the recovery configuration, choose **Back**.

11. If the settings are correct, choose **Finish**.

   The recovery is then started.

   The progress of the recovery for each SAP HANA service is displayed in the dialog box. When the recovery is complete, a message confirms this.
The SAP HANA database is online.

13.12 Copying a Database Using Backup and Recovery

You can create a homogenous copy of a database by recovering an existing source database backup to a different, but compatible target database. The source database backup consists of data backup files and the log backup files.

A homogenous database copy is a quick way to set up cloned systems for training, testing, and development. For this reason, copying a database can significantly reduce total cost of delivery (TCD).

**Note:** A database copy is currently only possible using file-based data backups.

The following prerequisites must be met:

- A database backup of the source database must be available.
- The version of the target database system is the same or higher than the source database.
- The target system must be configured with sufficient disk space and memory.
- Ensure that the configuration of the target system is identical to the configuration of the source system.

The number of hosts and the number and types of services (for example, index server) on each host must be identical for both system landscapes.

**Note:**
For information about copying a SAP HANA database from multiple hosts to a single host:

*Note 1749467 - Copying SAP HANA From a Multiple- to a Single-Host System*

- Customer-specific configuration changes can be manually applied to the target system.

**More information:** About the SAP HANA Configuration Files

**Note:**
Do not change any entries in nameserver.ini, section landscape. Do not change any entries in sapprofile.ini.

- Ensure that a license key file is available for the target database.

**Note:**
The old license key will not work in the target system.

To create a homogenous copy of a database:

1. Shut down the target database.
2. Move or delete available data and log backup files from the target database.
3. Copy the data backup files and, optionally, the log backup files from the source database to the corresponding directories in the target database.
4. Start the recovery of the target database system.
   In SAP HANA studio, select the target database system and choose Recovery... from the context menu.
5. Select the recovery type Recover the database to its most recent state.
6. Choose Next.
7. Choose Add to specify the location of the data and log backup files for the target database.
8. Select the option Initialize log area.
9. Confirm the warning.
10. Select the option Install new license key and specify the license key file.
11. Choose Next.
12. If the recovery options are correct, choose Finish. The recovery is then started.
   When the recovery is successfully completed, the database is started.

The source database has now been recovered to the target database.

You can now work with the recovered SAP HANA database.

13.13 Backup Checks

The following checks warn you of errors that are connected with the backup process:

<table>
<thead>
<tr>
<th>Check:</th>
<th>Check availability of volumes for backup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alert ID:</td>
<td>34</td>
</tr>
<tr>
<td>Description:</td>
<td>Before a backup is performed, this check warns you if the backup cannot be performed because a volume is unavailable.</td>
</tr>
<tr>
<td></td>
<td>This alert can be triggered in combination with the alerts NOT_ASSIGNED_VOLUMES and CHECK_INACTIVE_SERVICES.</td>
</tr>
<tr>
<td>Alert Text:</td>
<td>A backup cannot be performed because volume &lt;&lt;no&gt;(&lt;service&gt;)&gt;&gt; is not available or because the service assigned is not in ACTIVE_STATE=YES.</td>
</tr>
<tr>
<td>User Action:</td>
<td>Find out why the volume is not available.</td>
</tr>
<tr>
<td>Default Interval:</td>
<td>1 hour</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Check:</th>
<th>Check whether a data backup exists</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alert ID:</td>
<td>35</td>
</tr>
</tbody>
</table>
### Check: Check whether a data backup exists

**Description:** Checks whether at least one data backup exists and warns you if no successful data backup is available for the instance. You are warned before any actual data loss occurs.

**Alert Text:** No data backup exists.

**User Action:** To ensure that your database can be recovered, perform a data backup as soon as possible.

**Default Interval:** 6 hours

This check is also performed when the database is started.

### Check: Check last data backup

**Alert ID:** 36

**Description:** Checks whether the last data backup was successful and warns you if the last data backup failed. If a scheduled backup fails, this check can help you prevent a situation from arising where no current backups are available.

**Alert Text:** The last data backup was not successful.

**User Action:** Check the reason why the last data backup was not successful, resolve the problem, and perform a new data backup as soon as possible.

**Default Interval:** 24 hours

### Check: Check the age of the last data backup

**Alert ID:** 37

**Description:** Checks the age of the last successful data backup, and provides different levels of notification if the last successful data backup is too old.

**Alert Text:** The last data backup is already `<days>` days old.

**User Action:** To reduce your downtime in a recovery situation, perform a data backup as soon as possible.

**Default Interval:** 24 hours

### Check: Check last log backups.

**Alert ID:** 38

**Description:** Checks whether the last log backups were successful, and provides information about a failed log backup for a service or volume.
Check: Check last log backups.

As log backups are performed automatically, this is the only way to notify users. For this reason, it is important that this check be performed frequently and with a high priority.

Note: Log backups are performed individually and independently for each volume. For this reason, log backups need to be checked for each volume.

Alert Text: The last log backup was not successful for volume <<no> (<service> at <host>:<port>)>.

User Action: Check the reason why a log backup was not successful and resolve the problem.

Default Interval: 15 minutes

More Information

Monitoring Alerts in the SAP HANA Database – Administration Guide on the SAP Help Portal

13.14 SQL Syntax for Backup and Recovery

In addition to performing backup and recovery operations using SAP HANA studio, you can also use SQL statements. This section describes the syntax for these statements.

Pre-Requisites

To execute the SQL statements for backup:

- The database must be online.
- The user must have the privilege `BACKUP ADMIN`.

To execute the SQL statements for recovery:

- The database must be offline.
- SQL statements are executed as user `<SID>adm`.

SQL Statements

The following SQL statements are used for backup:

- `<execute_backup_data>`
The following SQL statements are used for recovery:

- `<execute_recover_data>`
- `<execute_recover_database>`
- `<backup_check_access>`
- `<backup_list_data>`

The following SQL statements are used for file system-based backup:

- `<execute_create_snapshot>`
- `<execute_drop_snapshot>`

### 13.14.1 SQL Statements for Data Backup

If no absolute path is specified for the data backup statement, the data backup is created in the default backup destination.

**Tip:**

To change the default destination for file-based data backups, in SAP HANA studio, go to the **Backup** section, then choose **Configuration** > **Data Backup Settings**.

Each complete data backup includes all the configured services (for example, name server, index server, and statistics server).

The name of a data backup consists of an optional prefix, the string `<databackup>`, the volume ID of the service and an index.

**Table 20: Syntax of SQL Statements for Data Backup**

<table>
<thead>
<tr>
<th>Statement</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;execute_backup_data&gt;</code></td>
<td>::= BACKUP DATA <code>&lt;data_file_definition&gt;</code> &lt;backup_option&gt;</td>
</tr>
<tr>
<td><code>&lt;data_file_definition&gt;</code></td>
<td>::= USING <code>&lt;file_definition&gt;</code></td>
</tr>
<tr>
<td><code>&lt;file_definition&gt;</code></td>
<td>::= FILE (&lt;file_name_prefix&gt;)</td>
</tr>
<tr>
<td><code>&lt;file_name_prefix&gt;</code></td>
<td>::= &lt;apostrophe&gt; <code>&lt;string_const&gt;</code> &lt;apostrophe&gt;</td>
</tr>
<tr>
<td><code>&lt;backup_option&gt;</code></td>
<td>::= ASYNCHRONOUS</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Examples of SQL Commands for Data Backup

- **BACKUP DATA USING FILE ('COMPLETE')**

Perform a data backup for the entire database and create a data backup for every relevant service in the default backup destination.

Data backups are prepended with the string `COMPLETE`.

The following files are created:

- `<default backup directory>/COMPLETE_databackup_0_1 (contains topology)`
- `<default backup directory>/COMPLETE_databackup_1_1 (contains name server persistency)`
- `<default backup directory>/COMPLETE_databackup_2_1 (normally contains index server persistency)`
- `<default backup directory>/COMPLETE_databackup_3_1 (normally contains statistic server persistency)`

- **BACKUP DATA USING FILE ('/backup/MONDAY')**

Perform a data backup for the entire database and create a data backup for every relevant service in the directory `/backup`.

Data backup files are prepended with the string `MONDAY`.

The following files are created:

- `/backup/MONDAY_databackup_0_1 (contains topology)`
- `/backup/MONDAY_databackup_1_1 (contains name server persistency)`
- `/backup/MONDAY_databackup_2_1 (normally contains index server persistency)`
- `/backup/MONDAY_databackup_3_1 (normally contains statistic server persistency)`

- **BACKUP DATA USING FILE ('/backup/') ASYNCHRONOUS**

Perform a data backup for the entire database and create a data backup file for every relevant service in the directory `/backup`.

Data backups are not labeled.
The `ASYNCHRONOUS` option returns to the client immediately after activating the backup operation, even though the data backup is still running.

The following backups are created:

- `/backup/databackup_0_1` (contains topology)
- `/backup/databackup_1_1` (contains name server persistency)
- `/backup/databackup_2_1` (normally contains index server persistency)
- `/backup/databackup_3_1` (normally contains statistic server persistency)

```
- BACKUP DATA USING BACKINT ('MyBackIntBackup') TOOLOPT 'This is my tooloption'
```

Perform a data backup for the entire instance and create a data backup for each service using the third-party backup tool.

Data backups are prepended with the string `MyBackIntBackup`.

The `TOOLOPT` string `'This is my tooloption'` is forwarded to the backup tool.

The following backups are created in the backup tool:

- `<default backup directory>/MyBackIntBackup_0_1` (contains topology)
- `<default backup directory>/MyBackIntBackup_1_1` (contains name server persistency)
- `<default backup directory>/MyBackIntBackup_2_1` (normally contains index server persistency)
- `<default backup directory>/MyBackIntBackup_3_1` (normally contains statistic server persistency)

### 13.14.2 SQL Statement for Data Backup Check

The following SQL statement checks whether enough space (in bytes) is available at the backup destination. The statement completes successfully or returns an error if the check predicts that the backup cannot be written due to insufficient space or if there are issues with permissions.

**Note:** This data backup check is only for file-based backups.

The result of this statement may vary as the free space in the file system is not controlled by the database and is constantly changing.

<table>
<thead>
<tr>
<th>Command</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;execute_backup_check&gt;</code></td>
<td><code>::= BACKUP CHECK &lt;data_file_definition&gt; SIZE &lt;size&gt;</code></td>
</tr>
<tr>
<td><code>&lt;data_file_definition&gt;</code></td>
<td><code>::= USING &lt;data_file&gt;</code></td>
</tr>
</tbody>
</table>
Example of an SQL Statement for Data Backup Check

```
BACKUP CHECK USING FILE ('/backup/data') SIZE 50000000000
```

13.14.3 SQL Statements for Data Backup Cancel

The following SQL statements cancel a running data backup that is identified by its unique backup ID.

If a backup cancelation is successful, any existing data backup fragments are removed automatically. A new data backup can be started afterwards.

If a backup cancelation fails, the backup will proceed until it is completed normally.

Table 22: Syntax of SQL Commands for Data Backup Cancel

<table>
<thead>
<tr>
<th>Command</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;execute_backup_cancel&gt;</code></td>
<td>::= <code>BACKUP CANCEL</code> <code>&lt;backup_ID&gt;</code></td>
</tr>
<tr>
<td><code>&lt;backup_ID&gt;</code></td>
<td>::= <code>&lt;integer&gt;</code></td>
</tr>
</tbody>
</table>

Example of an SQL Command for Data Backup Cancel

```
BACKUP CANCEL 1331715084250
```

More Information

**Canceling a Running Data Backup**

13.14.4 SQL Statements for Data Snapshot

SAP HANA database provides the capability to clone an entire database while online.

Cloning is done using the underlying storage system. For consistency, a database-wide snapshot of the data area is used, similar to the procedure for data backup.

After the database has been cloned, the snapshot is removed from the source database. The snapshot in the cloned database is restored during the first restart.

Then, the snapshot in the destination database is removed.
Table 23: Syntax of SQL Statements for Data Snapshot

<table>
<thead>
<tr>
<th>Command</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;execute_create_snapshot&gt;</td>
<td>::= BACKUP DATA CREATE SNAPSHOT</td>
</tr>
<tr>
<td>&lt;execute_drop_snapshot&gt;</td>
<td>::= BACKUP DATA DROP SNAPSHOT</td>
</tr>
</tbody>
</table>

Examples of SQL Statements for Data Snapshot

- BACKUP DATA CREATE SNAPSHOT
  Create a database-wide snapshot based on a transactional consistent savepoint similar to the data backup.
  If a snapshot exists, no complete data backup is possible.
  Every request of this kind is rejected with a notification that a data backup is still in process.
- BACKUP DATA DROP SNAPSHOT
  Drop a database-wide snapshot. From this point in time on, complete data backups are possible.

13.14.5 SQL Statements for Database Recovery and Data Recovery

This section describes the syntax of SQL statements for database recovery (data and log) and data recovery (data only, no log recovery).

Note: To recover a SAP HANA database, it is **strongly recommended that you use SAP HANA studio**. Statements for recovery cannot be executed using the normal SQL clients and should only be used in exceptional circumstances.

Database recovery is divided into the following steps:

1. Determine the recovery strategy.
   The recovery strategy is determined based on the most recent complete data backup in the backup catalog. If the most recent data backup is not used for the recovery, an older backup can be selected.
   When the recovery strategy has been determined, the recover statement is passed to the database.

2. Execute the specified and confirmed strategy.

Table 24: Syntax of Commands for Database Recovery and Data Recovery and Backint

<table>
<thead>
<tr>
<th>Command</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;execute_recover_database&gt;</td>
<td>::= RECOVER DATABASE &lt;until_definition&gt;</td>
</tr>
<tr>
<td></td>
<td>[&lt;source_definition&gt;] [USING BACKUP_ID</td>
</tr>
<tr>
<td></td>
<td>&lt;int_const&gt;] [USING GUID &lt;str_const&gt;]</td>
</tr>
<tr>
<td></td>
<td>[CHECK ACCESS [ALL</td>
</tr>
<tr>
<td></td>
<td>BACKINT]]</td>
</tr>
<tr>
<td>&lt;execute_recover_data&gt;</td>
<td>::= &lt;execute_recover_normal&gt;</td>
</tr>
</tbody>
</table>
### Command Syntax

<table>
<thead>
<tr>
<th>Command</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;execute_recover_normal&gt;</td>
<td>::= RECOVER DATA &lt;data_file_definition&gt; &lt;recover_normal_option&gt;</td>
</tr>
<tr>
<td>&lt;execute_recover_legacy&gt;</td>
<td>::= RECOVER DATA &lt;data_file_definition&gt; [recover_option]</td>
</tr>
</tbody>
</table>

**Note:**

In log mode legacy, redo logs are not backed up and can therefore not be recovered.

The `<execute_recover_legacy>` statement can be used only when the log mode is **legacy** or **overwrite**.

| <until_definition>                          | ::= UNTIL_TIMESTAMP <timestamp> [recover_option]         |
|                                             | | UNTIL LOG POSITION <int_const> AT VOLUME <int_const> [recover_option] |
| <timestamp>                                 | ::= 'YYYY-MM-DD HH:MM:SS'                               |
| <source_definition>                         | ::= <data_source_definition> [log_source_definition]    |
|                                             | | <log_source_definition> [data_source_definition]     |
| <data_source_definition>                   | ::= USING DATA PATH ( <source_path_list> )            |
| <log_source_definition>                    | ::= USING LOG PATH ( <source_path_list> )             |
| <source_path_list>                         | ::= <source_path> | <source_path_list> comma <source_path>             |
| <source_path>                              | ::= <apostrophe> <string> <apostrophe>                |
| <data_file_definition>                    | ::= USING <file_definition>                           |
| <file_definition>                          | ::= FILE ( <file_name_prefix> )                        |
|                                             | | BACKINT ( <file_name_prefix> )                       |
| <file_name_prefix>                         | ::= <apostrophe> <string_const> <apostrophe>          |
| <recover_option>                           | ::= <recover_option_clearlog>                         |
### Examples of Commands for Database Recovery and Data Recovery

**Example Scenario 1**

The last complete data backup was created on Monday 22nd at 8 am CEST (6am UTC).

The database ran up to 6pm CEST (4 pm UTC) and must be recovered up to 5pm (3 pm UTC).

Use the following command:

```sql
RECOVER DATABASE UNTIL TIMESTAMP '2011-08-22 15:00:00'
```

**Note:**

The timestamp given to the database is interpreted as UTC time. Therefore, 3pm means UTC and not 3pm CEST.

**Note:**

Note that the specified timestamp is not included in the transactions to be redone. For that reason, only transactions with a commit timestamp before the target timestamp will be redone.

**Example Scenario 2**

Recover the database until a point in time with a timestamp using a dedicated directory for data backups and further directories containing automatically written log backups.

Use the following command:
RECOVER DATABASE UNTIL TIMESTAMP '2011-08-22 15:00:00' USING DATA PATH ('/backup/MONDAY/') USING LOG PATH ('/backup/logs1/','/backup/logs2/')

- **Example Scenario 3**
  
  Perform data recovery for the entire database without replaying the redo log. The data backup files will be taken from the specified directory /backup.
  
  Data backups have to be labeled with prefix **MONDAY**.

  RECOVER DATA USING FILE ('/backup/MONDAY') CLEAR LOG

### Check Backup for Recovery

Checks whether the data backup specified by backup_id exists and can be accessed.

**Table 25: Syntax**

<table>
<thead>
<tr>
<th>Command</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;backup_check_access&gt;</td>
<td>::= BACKUP CHECK ACCESS BACKUP_ID &lt;int_const&gt; [source_definition]</td>
</tr>
</tbody>
</table>

If the option `<using path>` is not specified, the data backup is searched for only in the location specified in the backup catalog.

If the option `<using path>` is specified, the data backup is searched for recursively in the path specified (not in the backup catalog).

### List Backups for Recovery

Returns a list of data backups from the backup catalog that correspond to `<until_specification>`.

**Table 26: Syntax**

<table>
<thead>
<tr>
<th>Command</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;backup_list_data&gt;</td>
<td>::= BACKUP LIST DATA UNTIL &lt;until_definition&gt; [LIMIT &lt;int_const&gt; [OFFSET &lt;int_const&gt;]]</td>
</tr>
</tbody>
</table>

**LIMIT <limit>** restricts the number of elements returned.

If **LIMIT <limit>** is not specified, a default value of 10 is used.

**OFFSET <offset>** defines the number of data backups to skip at the beginning of the list.

If **OFFSET <offset>** is not specified, no data backups are skipped.
14 High Availability and SAP HANA

14.1 About High Availability

SAP HANA is fully designed for high availability. It supports recovery measures ranging from faults and software errors, to disasters that decommission an entire data center. High availability is the name given to a set of techniques, engineering practices and design principles that support the goal of business continuity. This is achieved by eliminating single points of failure (fault tolerance), and providing the ability to rapidly resume operations after a system outage with minimal business loss (fault resilience). Fault recovery is the process of recovering and resuming operations after an outage due to a fault. Disaster recovery is the process of recovering operations after an outage due to a prolonged data center or site failure. Preparing for disasters may require backing up data across longer distances, and may thus be more complex and costly.

The key to achieving High Availability is redundancy, including hardware redundancy, network redundancy and data center redundancy. SAP HANA provides several levels of defense against failure-related outages:

1. Hardware Redundancy – SAP HANA appliance vendors offer multiple layers of redundant hardware, software and network components, such as redundant power supplies and fans, enterprise grade error-correcting memories, fully redundant network switches and routers, and uninterrupted power supply (UPS). Disk storage systems use batteries to guarantee writing even in the presence of power failure, and use striping and mirroring to provide redundancy for automatic recovery from disk failures. Generally speaking, all these redundancy solutions are transparent to SAP HANA’s operation, but they form part of the defense against system outage due to single component failures.

2. Software – SAP HANA appliances use a carefully hardened and secure version of the Linux host operating system, using only certified drivers. Additionally, the SAP HANA system software also includes a watchdog function, which automatically restarts configured services (index server, name server, and so on), in case of detected stoppage (killed or crashed).

3. Persistence – SAP HANA persists transaction logs, savepoints and snapshots to support system restart and recovery from host failures, with minimal delay and without loss of data.

4. Standby and Failover – Separate, dedicated standby hosts are used for failover, in case of failure of the primary, active hosts. This improves the availability by significantly reducing the recovery time from an outage.

SAP HANA High Availability Support

As an in-memory database, SAP HANA is not only concerned with maintaining the reliability of its data in the event of failures, but also with resuming operations with most of that data loaded back in memory as quickly as possible.

SAP HANA supports the following recovery measures from failures:

- Disaster recovery support:
  - Backups: Periodic saving of database copies in safe place.
  - Storage replication: Continuous replication (mirroring) between primary storage and backup storage over a network (may be synchronous).
○ System replication: Continuous (currently: synchronous) update of secondary system by primary system, including in-memory table loading (formerly referred to as warm standby).

- Fault recovery support:
  ○ Service auto-restart: Automatic restart of stopped services on host (watchdog).
  ○ Host auto-failover: Automatic failover from crashed host to standby host in the same system.

14.1.1 SAP HANA Disaster Recovery Support

Backups

The SAP HANA database holds the bulk of its data in memory for maximum performance, but still uses persistent storage to provide a fallback in case of failure. During normal database operation, data is automatically saved from memory to disk at regular savepoints. A savepoint is a periodic point in time, when all the changed data is written to storage, in the form of pages. Additionally, all data changes are captured in the redo log. With each committed database transaction, the redo log is saved from memory to disk. If a failure occurs, for example a power outage, the database can be restarted in the same way as any disk-based database, and it is returned to its last consistent state by replaying the redo log since the last savepoint. Using data and log backups (and if available log entries from the log volumes), the SAP HANA database can be recovered.

See the section on Backing Up and Recovering the SAP HANA Database.

Storage Replication

One drawback of backups is the potential loss of data between the time of the last backup and the time of the failure. A preferred solution therefore, is to provide continuous replication of all persisted data. Several SAP HANA hardware partners offer a storage-level replication solution, which delivers a backup of the volumes or file-system to a remote, networked storage system. In some of these vendor-specific solutions, which are certified by SAP, the SAP HANA transaction only completes when the locally persisted transaction log has been replicated remotely. This is called synchronous storage replication. Synchronous storage replication can be used only where the distance between the primary and backup site is relatively short (typically 100 kilometers or less), allowing for sub-millisecond round-trip latencies. Due to its continuous nature, storage replication (sometimes also called remote storage mirroring) can be a more attractive option than backups, as it reduces the amount of time between the last backup and a failure. Another advantage of storage replication is that it also enables a much shorter recovery time. This solution requires a reliable, high bandwidth and low latency connection between the primary site and the secondary site.

See SAP Note 1755396 - Released DT solutions for SAP HANA with disk replication

System Replication

System replication employs an "N+N" approach, with a secondary standby system that is configured as an exact copy of the active, primary system. Each service instance of the primary SAP HANA system communicates with a counterpart in the secondary system. The secondary system can be located near the primary system to serve as...
a rapid failover solution for planned downtime, or to handle storage corruption or other local faults, or, it can be installed in a remote site to be used in a disaster recovery scenario. Like storage replication, this disaster recovery option requires a reliable link between the primary and secondary sites. The instances in the secondary system operate in recovery mode. In this mode, all secondary system services constantly communicate with their primary counterparts, replicate and persist data and logs, and load data to memory. The main difference is that the secondary system does not accept requests or queries.

When the secondary system is started in recovery mode, each service component establishes a connection with its counterpart, and requests a snapshot of the data in the primary system. From then on, all logged changes in the primary system are replicated. Whenever logs are persisted in the primary system, they are also sent to the secondary system. A transaction in the primary system is not committed until the logs are replicated. What this means in detail, can be configured by choosing one of the log replication modes:

- Synchronous: The primary system waits with committing the transaction until it receives a reply that the log is persisted in the secondary system. This mode guarantees immediate consistency between both systems, at a cost of delaying the transaction by the time for data transmission and persisting in the secondary system.
- Synchronous in-memory: The primary system commits the transaction after it receives a reply that the log was received by the secondary system, but before it was persisted. The transaction delay in the primary system is shorter, because it only includes the data transmission time.

If the connection to the secondary system is lost, or the secondary system crashes, the primary system after a brief, configurable, timeout will resume operations. The secondary system persists, but does not immediately replay the received log. To avoid a growing list of logs, incremental data snapshots are transmitted asynchronously from time to time from the primary system to the secondary system. If the secondary system has to take over, only that part of the log needs to be replayed that represents changes that were made after the most recent data snapshot. In addition to snapshots, the primary system also transfers status information regarding which table columns are currently loaded into memory. The secondary system correspondingly preloads these columns. In the event of a failure that justifies full system failover, an administrator instructs the secondary system to switch from recovery mode to full operation. The secondary system, which already preloaded the same column data as the primary system, becomes the primary system by replaying the last transaction logs, and then starts to accept queries.

**Related Links**
*Setting up System Replication*[page 199]

### 14.1.2 SAP HANA Fault Recovery Support

**Service Auto-Restart**

In the event of a software failure or an intentional intervention by an administrator, that disables one of the configured SAP HANA services (Index Server, Name Server, and so on), the service will be restarted by the SAP HANA service auto-restart watchdog function, which automatically detects the failure and restarts the stopped service process. Upon restart, the service loads data into memory and resumes its function. While all data remains safe the service recovery takes some time.
Host Auto-Failover

Host auto-failover is a local "N+m" (m is often 1) fault recovery solution that can be used in addition or as an alternative measure to system replication. One (or more) standby hosts are added to a SAP HANA system, and configured to work in standby mode. As long as they are in standby mode the databases on these hosts do not contain any data and do not accept requests or queries. This means they cannot be used for other purposes such as quality or test systems.

When a primary (worker) host fails, a standby host automatically takes its place. Since the standby host may take over operation from any of the primary hosts, it needs shared access to all the database volumes. This can be accomplished by a shared, networked storage server, by using a distributed file system, or with vendor-specific solutions that use a SAP HANA programmatic interface to dynamically detach and attach (mount) networked storage upon failover.

In support of host auto-failover, database clients can be configured with the connection information of multiple hosts, optionally including the standby host. The client connection code (ODBC, JDBC, and so on) will try to connect to one of these, and upon successful connection receives the updated connection configurations. This ensures that clients can continue to reach the SAP HANA database, even after failover.

Related Links

Setting up Host Auto-Failover [page 207]

14.2 Setting up System Replication

Prerequisites:

- The primary and secondary system are both installed and configured. You have verified that both are independently up and running.
- The number of hosts in the secondary system has to be equal to the number of hosts in the primary system.
- All configuration steps have to be executed on the master name server node only.
- The software version of the secondary has to be equal or newer to the version on the primary.
- The secondary system must have the same SAP system ID, <SID> and instance number as the primary system.

The primary replicates all relevant license information to the secondary. Once the SID is the same an additional license is not required.

- System replication between two systems on the same host is not supported.
- The .ini file configuration must be identical for both systems. Any changes made manually, or by SQL commands on one system must be manually duplicated on the other system.
- The required ports must be available. Typically they are set up as <instance number> + 1. Therefore no other SAP HANA system can use this port number.
The following configuration tasks need to be carried out on the primary and secondary systems to set up system replication. With this configuration you have the possibility to recover from a data center outage by switching to a secondary site.

1. Configuring the Primary System
2. Configuring the Secondary System

Related Links
Network Configuration [page 128]
SAP HANA Security Guide

14.2.1 Configuring the Primary System

- You are logged on to the master name server’s host as the root user.
- The user ID of the <sid>adm on the master system is free and can be used.
- If the hostnames of the primary and the secondary system are the same (for example, because two appliances are used that have identical hostnames) use hdbrename to change the hostnames used on the secondary system (as root user):

```shell
cd /usr/sap/<sid>/HDB<instancenr>/exe
./hdbrename -H <oldhost>=<newhost>
```

To set up system replication, the following configuration steps need to be carried out on the primary system.

This is done using the tool hdbnsutil, which initializes the topology of a database during the installation or exports, imports and converts the topology of an existing database and using the SAP HANA studio.

1. On the command line use hostname to check that the hostnames in the primary system are different from the hostnames used in the secondary system.
2. In the Administration editor of SAP HANA studio, choose the Configuration tab and ensure that log_mode is set to “normal” in the persistence section of the global.ini file.
   
   Log mode normal, means that log segments must be backed up. Log mode overwrite means log segments are freed by the savepoint (therefore only useful for test installations without backup and recovery).
3. Do an initial data backup.
4. As <sid>adm on the command line enable system replication with the following command. The primary system must be online at this time:

```shell
cd /usr/sap/<sid>/HDB<instancenr>/exe
./hdbnsutil -sr_enable --name=<primary_alias>
```

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--name</td>
<td>&lt;primary_alias&gt;</td>
<td>Alias used to represent your primary site and assign it as the primary site for system replication</td>
</tr>
</tbody>
</table>
- used to give a name to your primary site and assign it as the primary site for system replication.

If the site has been successful enabled for system replication you will see the following output:

```
successfully enabled system as primary site ...
done.
```

Now you can config the secondary system.

Related Links

Performing a Data Backup Using SAP HANA Studio [page 176]
Configuring the Secondary System [page 201]

14.2.2 Configuring the Secondary System

- You have already configured a primary system, so that you can register a secondary system with it.
- You are logged on to the master name server’s host as the root user.
- The user ID of the `<sid>adm` on the master system is free and can be used.
- If the hostnames are the same (for example, because two appliances are used that have identical hostnames) use `hdbrename` to change the hostnames used on the secondary system (as root user):

```
cd /usr/sap/<sid>/HDB<instancenr>/exe

./hdbrename -H <oldhost>=<newhost>
```

There are two mode of log replication that can be used to send log information to the secondary instance. You need to decide which mode to use.

- Synchronous (mode=sync) means the log write is considered as successful when the log entry has been written to the log file of the primary and the secondary instance. No data loss occurs in this scenario.
- Synchronous in memory (mode=syncmem) means the log write is considered as successful, when the log entry has been written to the log file of the primary and sending the log has been acknowledged by the secondary instance after copying to memory. Data loss can occur when primary and secondary fail at the same time. This option provide better performance, because it is not necessary to wait for disk I/O on the secondary instance, but is more vulnerable to data loss.

To set up system replication, the following configuration steps need to be carried out on the secondary system.

This is done using the tool `hdbnsutil`, which initializes the topology of a database during the installation or exports, imports and converts the topology of an existing database and using the SAP HANA studio.

1. On the command line use `hostname` to check that the hostnames in the primary system are different from the hostnames used in the secondary system.
2. In the Administration editor of SAP HANA studio, choose the Configuration tab and ensure that log_mode is set to “normal” in the persistence section of the `global.ini` file.

Log mode normal, means that log segments must be backed up. Log mode overwrite means log segments are freed by the savepoint (therefore only useful for test installations without backup and recovery).
3. Log on to the secondary system as a user with root authorization and use the `sapcontrol` program to execute the following command to shut down the system:

```
/usr/sap/hostctrl/exe/sapcontrol -nr <Instance_Number> -function Stop
```

4. Enable system replication on the secondary system with the following command:

```
hdbnsutil -sr_register --name=<secondary_alias> --remoteHost=<primary_host> --remoteInstance=<primary_systemnr> --mode=[sync|syncmem]
```

Table 27: hdbnsutil Call Options

<table>
<thead>
<tr>
<th>Option Name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--name</td>
<td>&lt;secondary_alias&gt;</td>
<td>Alias used to represent the secondary site</td>
</tr>
<tr>
<td>--remoteHost</td>
<td>&lt;primary_host&gt;</td>
<td>Name of the primary host that the secondary registers with</td>
</tr>
<tr>
<td>--remoteInstance</td>
<td>&lt;primaryinstancenr&gt;</td>
<td>Instance number of primary</td>
</tr>
<tr>
<td>--mode</td>
<td>[sync</td>
<td>syncmem]</td>
</tr>
</tbody>
</table>

If you have successfully registered the secondary with the primary you will see this output:

```
nameserver <hostname><port> not responding # ok, as server needs to be offline for the registration
collecting information ...
sending registration request to primary site host <hostname><port>
updating local ini files ...
nameserver <hostname><port> not responding. # As comment above done.
```

5. Start the secondary system in order to reinitialize it with the following command:

```
cd /usr/sap/<sid>/HDB<instancenr>
./HDB start
```

You have registered the secondary system with the primary system and system replication is enabled.

Related Links

*Monitoring System Replication* [page 205]

### 14.2.3 Performing a Takeover

If a disaster occurs and the primary data center is no longer available and a decision has made to fail over to the secondary data center the procedure described here can be used.

The takeover command can be executed both when the secondary system is in an offline state or online state. If the secondary system is online, it can be brought out of recovery mode and become fully operational, meaning it now accepts and responds to queries, using the following steps:
As <sid>adm enter the following command to enable the secondary system to take over and become the primary system:

```
cd /usr/sap/<sid>/HDB<instancenr>/exe
./hdbnsutil -sr_takeover
```

If the system is offline, the takeover is actually carried out when the system is next started.

### 14.2.4 Performing a Failback

After a takeover has been carried out, and the data center is back in operation the roles between primary and secondary can be switched over. In this case, the former primary now has to be registered as the secondary with the active primary system.

This is the same procedure as is used for setting up a normal secondary described in *Configuring the Secondary System*. The data, that is already available in the persistence cannot be reused, instead a complete initialization will be carried out. This means a full replica will be shipped until the "old primary" is in sync again.

**Related Links**

*Configuring the Secondary System* [page 201]

### 14.2.5 Example of Setting up System Replication

Two identical SAP HANA appliances are shipped, both with the default hostname, hananode01. To set up system replication with two appliances you have to change the hostnames.

In this example a single node system is used, in multi-node systems all hosts have to renamed.

**Note:** To use the command line program `hdbrename` to rename hosts in a production system replication landscape, system replication must be first deactivated. This means you have to first unregister and disable the secondary system before renaming any hosts. Once you have renamed the hosts then you can enable recovery mode again and register the secondary system with the primary system to re-activate system replication.

1. As <sid>adm open a command line on the respective host and use `hdbrename` to change the hostname.
   a) On the primary system change the hostname to "ej11" with the command:
```
cd /usr/sap/<sid>/HDB<instancenr>/exe
./hdbrename -H hananode01=ej11
```
   b) On the secondary system change the hostname to "nh19" with the command:
```
cd /usr/sap/<sid>/HDB<instancenr>/exe
./hdbrename -H hananode01=nh19
```
2. Enable system replication on the primary system: ej11.

   cd /usr/sap/<sid>/HDB<instancenr>/exe
   ./hdbnsutil -sr_enable --name=dcsite1

3. Stop the secondary system with HDB stop. The primary system can stay online.

4. Register the secondary system with the following command:

   cd /usr/sap/<sid>/HDB<instancenr>/exe
   ./hdbnsutil -sr_register
   --name=dcsite2
   --remoteHost=ej11
   --remoteInstance=50
   --mode=sync

Related Links
See SAP Note 611361 - Hostnames of SAP servers.

14.2.6 System Replication Configuration Parameters

All parameters described in this section are defined in the system replication section of the global.ini file and have the default values shown below. The column System defines whether the parameter can be set on the primary, the secondary or both.

1. Note: preload_column_tables uses the boolean keywords “true” or “false”. Numbers do not work in place of the keywords.

Table 28: Configuration Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Unit</th>
<th>Default</th>
<th>System</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>datashipping_min_time_interval</td>
<td>int</td>
<td>seconds</td>
<td>600 (10 min)</td>
<td>Secondary</td>
<td>Minimum time interval between two data shipping requests from secondary system. If datashipping_logsize_threshold (see next parameter) is reached first, the data shipping request will be sent before the time interval is elapsed, when the log size threshold is reached.</td>
</tr>
<tr>
<td>datashipping_logsize_threshold</td>
<td>int</td>
<td>bytes</td>
<td>5<em>1024</em>1024*10 24 (5GB)</td>
<td>Secondary</td>
<td>Minimum amount of log shipped between two data shipping requests from secondary system. If the time defined by datashipping_min_time_interval (see previous parameter) has passed before reaching this threshold, the data shipping request will be sent before this threshold is reached, when the time interval has elapsed.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Unit</td>
<td>Default</td>
<td>System</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------</td>
<td>------------</td>
<td>---------</td>
<td>--------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>preload_column_tables</td>
<td>bool</td>
<td>(true/</td>
<td>true</td>
<td>Both</td>
<td>If set, preload of column table main parts is activated. If set on the primary system, the loaded table information is collected and stored in the snapshot that is shipped. If set on the secondary system, this information is evaluated and the tables are actually preloaded there according to the information received on the loaded tables.</td>
</tr>
<tr>
<td>logshipping_timeout</td>
<td>int</td>
<td>seconds</td>
<td>30</td>
<td>Primary</td>
<td>Number of seconds, the primary system waits before shipping of a single log buffer. If the log shipping request is not handled within the configured time frame, it is assumed, that an error situation occurred. In this case the log buffer is freed and the replication session is canceled.</td>
</tr>
<tr>
<td>reconnect_time_interval</td>
<td>int</td>
<td>seconds</td>
<td>30</td>
<td>Secondary</td>
<td>If a secondary system is disconnected from the primary system due to network problems, the secondary tries to reconnect periodically after the time interval specified in this parameter has passed.</td>
</tr>
</tbody>
</table>

### 14.2.7 Monitoring System Replication

The primary system provides a monitoring view, M_SERVICE_REPLICATION, that contains the basic status of replication between primary and secondary systems.

Because the secondary instance does not accept SQL connections while data replication is active, basic information about the secondary system is also provided by this monitoring view on the primary site.

You can monitor the general status of the secondary system in the Overview tab of the Administration editor in SAP HANA studio. The Landscape tab provides a subtab called Secondary System Status which contains detailed information.

Log Shipping Timeouts generate alerts. You can receive these alerts by email.

**Related Links**

*Monitoring Overall System Status* [page 62]

When you open the Administration editor for a particular SAP HANA system, the Overview tab provides you with a summary of the overall status of the system, as well as an overview of resource usage.

*Monitoring Alerts* [page 73]

As the administrator of an SAP HANA system, you actively monitor the status of the system, its services, and the consumption of system resources. However, you are also alerted of critical situations, for example: a disk is becoming full, CPU usage is reaching a critical level, or a server has stopped.

*Configuring E-Mail Notifications for Alerts* [page 82]
You can configure the statistics server in such a way that you receive an e-mail when an alert condition for all or specific checks is fulfilled.

SAP HANA System Tables and Monitoring Views Reference

14.2.8 Adding a New Host to a Replicated System

Adding a new host to a system, that is currently being replicated is not automatically supported. It is possible to add a new host by manually performing the following steps (as <sid>adm on the command line):

1. **Note:** Hosts must be added equally to both primary and secondary sites.

   1. Shutdown the secondary system

      ```
      cd /usr/sap/<sid>/HDB<instancenr>
      ./HDB stop
      ```

   2. Unregister the secondary system

      ```
      cd /usr/sap/<sid>/HDB<instancenr>/exe
      ./hdbnsutil -sr_unregister
      ```

   3. Add the new host to the primary system (the primary system is still running):

      ```
      cd /usr/sap/<sid>/SYS/global/hdb/install/bin
      ./hdbaddhost ...
      ```

   4. Start the secondary system

      ```
      cd /usr/sap/<sid>/HDB<instancenr>
      ./HDB start
      ```

   5. Add a new host to secondary system:

      ```
      cd /usr/sap/<sid>/SYS/global/hdb/install/bin
      ./hdbaddhost ...
      ```

   6. Shutdown the secondary system

      ```
      cd /usr/sap/<sid>/HDB<instancenr>
      ./HDB stop
      ```
7. Re-register the secondary system

```
    cd /usr/sap/<sid>/HDB<instancenr>/exe
    ./hdbnsutil -sr_register ...
```

8. Start up the secondary system again.

```
    cd /usr/sap/<sid>/HDB<instancenr>
    ./HDB start
```

### 14.3 Setting up Host Auto-Failover

To add an additional standby host follow the procedure as described in *Adding a Host*.

When following this procedure you specify the role as "standby" to create a standby host.

You can monitor the status of all active and standby hosts in the Administration editor.

**Related Links**

- Perform a Distributed SAP HANA Setup [page 130]
- Monitoring in the Administration Editor [page 61]
  - To identify problems with the SAP HANA database early and avoid disruptions, you need to monitor your systems continuously.
- Configuring Clients for Failover [page 207]

### 14.3.1 Configuring Clients for Failover

To support failover with client libraries you have to specify a list of hostnames separated by a semicolon instead of a single host name. The client will choose one of these hosts to connect to. If a host is not available the next host from the list will be used. Only in the case that none of the hosts are available will you get a connection error.

If a connection gets lost when a host isn’t available any longer the client will reconnect to one of the host specified in the host list.

**Table 29: Example Configurations**

<table>
<thead>
<tr>
<th>Client</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>JDBC</td>
<td>Connect URL: jdbc:sap://host1:30015;host2:30015;host3:30015/</td>
</tr>
</tbody>
</table>
| SQLDBC  | SQLDBC_Connection *conn = env.createConnection();
         |   SQLDBC_Retcode rc = conn->connect("host1:30015;host2:30015;host3:30015", ",
<pre><code>     |   &quot;user&quot;, &quot;password&quot;);                                               |
</code></pre>
<table>
<thead>
<tr>
<th>Client</th>
<th>Example</th>
</tr>
</thead>
</table>
| ODBC   | Connect URL:  
"DRIVER=HDBODBC32;UID=user;PWD=password;SERVERNODE=host1:30015;host2:30015;host3:30015;" |
15 Query Analysis in the SAP HANA Database

15.1 Analyzing SQL Execution with the Plan Explanation

You can generate a plan explanation for any SQL statement in the SQL console. You can use this to evaluate the execution plan that the SAP HANA database follows to execute an SQL statement.

1. Enter a query into the SQL console and choose Explain Plan in the context menu.

   **Note:** You can enter multiple statements, separated by the configured separator character (usually a semicolon), to generate several plan explanations at once.

   The plan explanation is displayed on the Result tab.

2. Optional: Run the same statement on different systems/users by changing the SQL connection.
   That is, assuming that the tables and views exist in the other systems and you have authorization to access them.

   The plan explanation is displayed on the Result tab.

Plan explanations are also stored in the EXPLAIN_PLAN_TABLE view of the SYS schema for later examination.

15.2 Analyzing SQL Execution with the Plan Visualizer

To help you understand and analyze the execution plan of an SQL statement, you can generate a graphical view of the plan.

1. Visualize the plan of the SQL statement in one of the following ways:
   a) Enter the statement in the SQL console and choose Visualize Plan in the context menu.
   b) On the SQL Plan Cache tab or the Expensive Statements Trace tab of the Performance tab, right-click the statement and choose Visualize Plan.

   A graphical representation of the query, with estimated performance, is displayed.

   **Figure 14: Visualized Plan**
2. Validate the estimated performance by choosing **Execute** in the context menu. Another similar high-level graphic is generated with execution time information for each of the parts.

*Figure 15: Executed Plan*
3. To see a temporal breakdown of the individual operations processed in the execution of the query, open the **Timeline** view.

From the main menu choose > **Window** > **Show View** > **Timeline**

*Figure 16: Timeline View*

**Note:** Execution time is given as a pair of values: "self" (the execution time of the node), and "Inclusive" (the execution time including the descendent nodes).
4. If the query used the SAP HANA Column Engine, you can view the details of the various database operations by choosing **Visualize Column Plan** from the context menu. A detailed graphic is displayed.

*Figure 17: Column Plan Visualization*
This graphic is a very powerful tool for studying performance of queries on SAP HANA databases. You can explore the graphic further, for example, you can expand, collapse, or rearrange nodes on the screen. You can also save the graphic as an image or XML file, for example, so you can submit it as part of a support query.

15.3 Analyzing SQL Execution with the SQL Plan Cache

The SQL plan cache is a valuable tool for understanding and analyzing SQL processing.

Before it is executed, every SQL statement is compiled to a plan. Once a plan has been compiled, it is better to reuse it the next time the same statement is executed, instead of compiling a new plan every time. The SAP HANA database provides an object, the SQL plan cache, that stores plans generated from previous executions. Whenever the execution of a statement is requested, a SQL procedure checks the SQL plan cache to see if there is a plan already compiled. If a match is found, the cached plan is reused. If not, the statement is compiled and the generated plan is cached.
As the SQL plan cache collects statistics on the preparation and execution of SQL statements, it is an important tool for understanding and analyzing SQL processing. For example, it can help you to find slow queries, as well as measure the overall performance of your system.

You can access the SQL plan cache in the Administration editor on the Performance tab. The two monitoring views associated with the SQL plan cache are M_SQL_PLAN_CACHE_OVERVIEW and M_SQL_PLAN_CACHE in the _SYS_STATISTICS schema.

The SQL plan cache contains a lot of information. Filtering according to the following columns can help you identify statements that are more likely to be causing problems and/or could be optimized:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL_EXECUTION_TIME</td>
<td>The total time spent for all executions of a plan</td>
</tr>
<tr>
<td></td>
<td>This helps to identify which statements are dominant in terms of time.</td>
</tr>
<tr>
<td>AVG_EXECUTION_TIME</td>
<td>The average time it takes to execute a plan execution</td>
</tr>
<tr>
<td></td>
<td>This can help you identify long-running SQL statements.</td>
</tr>
<tr>
<td>EXECUTION_COUNT</td>
<td>The number of times a plan has been executed</td>
</tr>
<tr>
<td></td>
<td>This can help you identify SQL statements that are executed more frequently than expected.</td>
</tr>
<tr>
<td>TOTAL_LOCK_WAIT_COUNT</td>
<td>The total number of waiting locks</td>
</tr>
<tr>
<td></td>
<td>This can help you identify SQL statements with high lock contention.</td>
</tr>
<tr>
<td>USER_NAME</td>
<td>The name of the user who prepared the plan and therefore where the SQL originated (ABAP/indexserver/statisticsserver)</td>
</tr>
</tbody>
</table>

**Related Links**

*Monitoring System Performance* [page 84]

In addition to the general information about the overall system performance that you can see in the System Monitor and on the Overview tab of the Administration editor, you can monitor more detailed aspects of system performance on the Performance tab.

*SAP HANA Database - System Tables and Monitoring Views*

**15.4 Example: Reading the SQL Plan Cache**

These examples aim to show you how to gain useful insights by analyzing the SQL plan cache.

**Execution in a Single-Host System**

This example aims to show you how to interpret information about execution time. The following table is sorted by TOTAL_EXECUTION_TIME.
You could read these top 3 results as follows:

- Statement 1 takes the longest time overall but it is also executed frequently.
- Statement 2 is not executed very frequently but has the second highest total execution time. Why is this simple SQL taking so long? Does it have problems processing?
- The execution times for statement 3 are fine for one-off execution, but it runs too frequently, over 20,000 times. Why? Is there a problem in application code?

Sorting by AVG_EXECUTION_TIME or EXECUTION_COUNT provides a different angle on your analysis.

This example aims to show you how to interpret information about locking situations. The information in columns TOTAL_LOCK_WAIT_COUNT and TOTAL_LOCK_WAIT_DURATION lets us know which statement is waiting for others and how much time it takes.

<table>
<thead>
<tr>
<th>USER_NAME</th>
<th>STATEMENT_STRING</th>
<th>TOTAL_LOCK_WAIT_COUNT</th>
<th>TOTAL_LOCK_WAIT_DURATION</th>
<th>TOTAL_EXECUTION_TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSTEM</td>
<td>SELECT &quot;FROMNUMBER&quot;,&quot;TONUMBER&quot;,&quot;NRLEVEL&quot; FROM &quot;NRIV&quot; ... FOR UPDATE</td>
<td>11,549,961</td>
<td>210,142,307,207</td>
<td>676,473</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>UPDATE &quot;NRIV&quot; SET &quot;NRLEVEL&quot; = ? WHERE &quot;CLIENT&quot; = '0000' ... AND &quot;TOYEAR&quot; = '0000'</td>
<td>0</td>
<td>0</td>
<td>3,706,184</td>
</tr>
</tbody>
</table>
Here, it is clear that the first statement is waiting almost all the time. If the application is critical, it may be necessary to revise the `SELECT...FOR UPDATE` code for better resource utilization and performance.

**Distributed Execution**

In distributed SAP HANA systems, tables and table partitions are located on multiple hosts. The execution of requests received from database clients may potentially have to be executed on multiple hosts, depending on where the requested data is located. The following example illustrates how, if statement routing is not enabled, requests from the database client are executed on the contacted index server (in this case the master index server) and the required data is fetched from the index server on the relevant host(s). However, if statement routing is enabled, after initial query compilation, request execution is routed directly to the host on which the required data is located.

![Distributed Execution with Statement Routing Off and On](image)

Execution times should be better with statement routing enabled. You can use the SQL plan cache to compare the execution statistics of statements with statement routing enabled and disabled and thus confirm the effect.

Statement routing is controlled by the `client_distribution_mode` parameter in the `indexserver.ini` file. It is enabled by default (value=statement).
The following SQL plan cache examples show the execution times of sample statements based on the scenario illustrated above.

Note: The column IS_DISTRIBUTED_EXECUTION indicates whether or not statement execution takes place on more than one host.

Statement Routing Off

<table>
<thead>
<tr>
<th>HOST</th>
<th>PORT</th>
<th>STATEMENT_ STRING</th>
<th>IS_DISTRIBUTED_EXECUTION</th>
<th>TABLE_LOCATIONS</th>
<th>EXECUTION_COUNT</th>
<th>TOTAL_EXECUTION_TIME</th>
<th>AVG_EXECUTION_TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>seltera12</td>
<td>33103</td>
<td>UPSERT &quot;RSBMONMESS_DTIP&quot; (&quot;MSGNO&quot;, &quot;MSGTY&quot;, &quot;MSGID&quot;, ...)</td>
<td>FALSE</td>
<td>(seltera12.33103, 2)</td>
<td>13</td>
<td>16792</td>
<td>1044</td>
</tr>
<tr>
<td>seltera12</td>
<td>33103</td>
<td>UPSERT &quot;RSBMONMESS_DTIP&quot; (&quot;MSGNO&quot;, &quot;MSGTY&quot;, &quot;MSGID&quot;, ...)</td>
<td>TRUE</td>
<td>(selbld13.33103.5)</td>
<td>91</td>
<td>125085</td>
<td>1374</td>
</tr>
<tr>
<td>seltera12</td>
<td>33103</td>
<td>SELECT * FROM &quot;/B49/SCUSTORE R&quot; WHERE &quot;&quot;</td>
<td>TRUE</td>
<td>(selbld13.33103.5)</td>
<td>12144</td>
<td>142623926</td>
<td>1174</td>
</tr>
</tbody>
</table>

Figure 19: Statement Routing On

<table>
<thead>
<tr>
<th>HOST</th>
<th>PORT</th>
<th>STATEMENT_ STRING</th>
<th>IS_DISTRIBUTED_EXECUTION</th>
<th>TABLE_LOCATIONS</th>
<th>EXECUTION_COUNT</th>
<th>TOTAL_EXECUTION_TIME</th>
<th>AVG_EXECUTION_TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>seltera12</td>
<td>33103</td>
<td>UPSERT &quot;RSBMONMESS_DTIP&quot; (&quot;MSGNO&quot;, &quot;MSGTY&quot;, &quot;MSGID&quot;, ...)</td>
<td>FALSE</td>
<td>(seltera12.33103.2)</td>
<td>20</td>
<td>3061773</td>
<td>153088</td>
</tr>
<tr>
<td>selbld13</td>
<td>33103</td>
<td>UPSERT &quot;RSBMONMESS_DTIP&quot; (&quot;MSGNO&quot;, &quot;MSGTY&quot;, &quot;MSGID&quot;, ...)</td>
<td>FALSE</td>
<td>(selbld13.33103.5)</td>
<td>18</td>
<td>124625</td>
<td>1153</td>
</tr>
<tr>
<td>selbld16</td>
<td>33103</td>
<td>SELECT * FROM &quot;/B49/SCUSTORE R&quot; WHERE &quot;&quot;</td>
<td>FALSE</td>
<td>(selbld16.33103.6)</td>
<td>83823</td>
<td>32611413</td>
<td>1174</td>
</tr>
</tbody>
</table>

The TOTAL_EXECUTION_TIME for a statement is the sum of execution times on all hosts, therefore:

Table 30: Statement Routing Off

<table>
<thead>
<tr>
<th>Statement</th>
<th>Request Path</th>
<th>Total Execution Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>UPSERT &quot;RSBMONMESS_DTIP&quot; (&quot;MSGNO&quot;, &quot;MSGTY&quot;, &quot;MSGID&quot;, ...)</td>
<td>seltera12</td>
<td>= execution time on seltera12</td>
</tr>
<tr>
<td>SELECT * FROM &quot;/BIO/SIOBJNM&quot; WHERE &quot;IOBJNM&quot; = ?</td>
<td>seltera12 &gt; selbld13</td>
<td>= execution time on seltera12 + execution time on selbld13</td>
</tr>
<tr>
<td>SELECT * FROM &quot;/B49/SCUSTOMER&quot; WHERE &quot;/B49/S_CUSTOMER&quot; = ?</td>
<td>seltera12 &gt; selbld13 &gt; selbld16</td>
<td>= execution time on seltera12 + execution time on selbld16</td>
</tr>
</tbody>
</table>
### Table 31: Statement Routing On

<table>
<thead>
<tr>
<th>Statement</th>
<th>Request Path</th>
<th>Total Execution Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>UPSERT &quot;RSBMONMESS_DTP&quot; (&quot;MSGNO&quot;, &quot;MSGTY&quot;, &quot;MSGID&quot;, ...)</td>
<td>seltera12</td>
<td>= execution time on seltera12</td>
</tr>
<tr>
<td>SELECT * FROM &quot;/BI0/SIOBJNM&quot; WHERE &quot;IOBJNM&quot; = ?</td>
<td>selbld13</td>
<td>= execution time on selbld13</td>
</tr>
<tr>
<td>SELECT * FROM &quot;/B49/SCUSTOMER&quot; WHERE &quot;/B49/S_CUSTOMER&quot; = ?</td>
<td>selbld16</td>
<td>= execution time on selbld16</td>
</tr>
</tbody>
</table>
16  Getting Support

16.1  Opening a Support Connection

In some support situations, it may be necessary to allow an SAP support engineer to log into your system to analyze the situation.

1. To enable a support user to log on to your system, complete the following tasks:
   a) Install the SAPRouter as described on SAP Support Portal.
   b) Set up a support connection as described in SAP Note 1634848 (SAP HANA database service connections).
   c) Configure a Telnet connection as described in SAP Note 37001 (Telnet link to customer systems).
   d) Configure SAP HANA database connection as described in SAP Note 1592925 (SAP HANA Studio service connection).
   e) Configure a TREX/BIA/HANA service connection as described in SAP Note 1058533 (TREX/BIA/HANA service connection to customer systems).

2. Create a database user and grant the MONITORING role.

The MONITORING role allows a database user to open the Administration Console perspective with read-only access to the system, monitoring views, statistics server, trace files, and so on. However, this role does not provide any privileges for accessing application data. With the MONITORING role, it is also not possible to change the configuration of or start and stop a system. You can grant the MONITORING role to a support engineer if SAP support needs to connect to the system. Depending on the issue to be analyzed, further privileges may be needed to allow sufficient analysis (for example, to access application data or data models).

Related Links
SAP Support Portal
SAP Note 1634848
SAP Note 37001
SAP Note 1592925
SAP Note 1058533

16.2  Diagnosis Files

Diagnosis files include log and trace files, as well as a mixture of other diagnosis, error, and information files. In the event of problems with the SAP HANA database, you can check these diagnosis files for errors.

You can access diagnosis files on the Diagnosis Files tab of the Administration editor. They are stored at the following default location: `\usr\sap\<SID>\HDB<instance>\<host>\trace`.

Recommendation: Monitor disk space that is used for diagnosis files and delete files that are no longer needed.

Related Links
Monitoring Overall System Status [page 62]
When you open the Administration editor for a particular SAP HANA system, the Overview tab provides you with a summary of the overall status of the system, as well as an overview of resource usage.

Monitoring Disk Space [page 88]
To ensure that the database can always be restored to its most recent committed state, you must ensure that there is enough space on disk for data and log volumes. You can monitor disk usage, volume size, and other disk activity statistics on the Volumes tab of the Administration editor.

16.3 Working with Diagnosis Files

In the event of problems with the SAP HANA database, you can display the relevant diagnosis file and analyze for errors. You can also filter, merge, delete, and download diagnosis files.

To be able view diagnosis files and delete trace files, you must have the system privilege TRACE ADMIN.

1. In the Administration editor, choose the Diagnosis Files tab. All diagnosis files are listed.
2. Perform the required action(s):

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter list</td>
<td>To refine the list of diagnosis files, specify a filter. For example, if you want to see only SQL trace files, enter sqltrace.</td>
</tr>
<tr>
<td>Display file</td>
<td>To display a file in the list, right-click it and choose Open, or double-click the file. The Show Start of File and Show End of File buttons help you to navigate particularly large files more easily. You can specify how many lines you want to see when you filter the file in this way.</td>
</tr>
<tr>
<td>Merge files</td>
<td>You can merge the diagnosis files listed on the Diagnosis Files tab by choosing Choose Merge Files…. This feature is helpful during troubleshooting as it allows you to review multiple diagnosis files of different file types at the same time. The merged file is created from the most recent diagnosis files. Once the file has been created, you can use the filtering options and timeframe slider to drill down and analyze further. Note: Merging diagnosis files can take a long time depending on the size and number of files to be merged.</td>
</tr>
<tr>
<td>Delete log files and other non-trace files (for example, *.log, *.tpt, *.py)</td>
<td>You can delete one or more individual files from the list by selecting the file(s) in question and in the context menu, choosing Delete.</td>
</tr>
<tr>
<td>Delete trace files (*trc)</td>
<td>There are two ways to delete trace files: ○ You can delete trace files in the same way as other diagnosis files by right-clicking them and choosing Delete. Note: The file may not actually be deleted. If a running service is currently writing to the file, it cannot be deleted. If this is the case, the file disappears from the list in the SAP HANA studio and is hidden in the file system at operating system level. As long as a service is still writing to the file, it still exists and consumes disk space. Once the file...</td>
</tr>
</tbody>
</table>
### Action Description

- **Reaches its maximum size, the system stops writing to it and creates a new trace file.**
- **Note:** You can batch delete trace files, for example all the trace files of a specific service, by choosing *Delete Trace Files* and making the required selection.
- **Download files**

| Download files | To download a diagnosis file for offline analysis, right-click and choose *Download*. You can select multiple files to download. |

### Related Links

*Configuring Trace File Rotation* [page 36]

Trace file rotation prevents trace files from growing indefinitely by limiting the size and number of trace files. You can configure trace file rotation globally for all services in the system and for individual services.

#### 16.4 Configuring Traces

You can activate and configure several traces on the *Trace Configuration* tab of the Administration editor. Trace data is saved to trace files, which you can view on the *Diagnosis Files* tab.

To be able to configure traces, you must have the system privilege TRACE ADMIN. To configure the kernel profiler, you must have the SUPPORT standard role and either the system privilege RESOURCE_ADMIN or TRACE_ADMIN.

You can configure the following traces:

<table>
<thead>
<tr>
<th>Trace</th>
<th>Default Configuration/Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQL trace</td>
<td>Inactive</td>
</tr>
<tr>
<td>Performance trace</td>
<td>Inactive</td>
</tr>
<tr>
<td>Kernel profiler trace</td>
<td>Inactive</td>
</tr>
<tr>
<td>Global database trace</td>
<td>Active with default trace level ERROR</td>
</tr>
<tr>
<td>Database trace</td>
<td>Active with default trace level ERROR</td>
</tr>
<tr>
<td>User-specific trace</td>
<td>Not specified</td>
</tr>
<tr>
<td>End-to-end trace</td>
<td>Active with default trace level ERROR</td>
</tr>
<tr>
<td>Expensive statements trace</td>
<td>Inactive</td>
</tr>
</tbody>
</table>

1. In the Administration editor, choose the *Trace Configuration* tab.
2. Choose the *Edit Configuration* button for the trace that you want to configure.
   The *Trace Configuration* dialog box appears.
3. Make the required settings.
   The configuration options available in the *Trace Configuration* dialog box depend on the trace type.
Note: To restore the default status or configuration of a trace, in the Trace Configuration dialog box choose Restore Defaults.

Related Links

Diagnosis Files [page 219]

Diagnosis files include log and trace files, as well as a mixture of other diagnosis, error, and information files. In the event of problems with the SAP HANA database, you can check these diagnosis files for errors.

16.4.1 SQL Trace

The SQL trace collects information about all executed SQL statements and saves it in a trace file for further analysis.

Information collected includes overall execution time of each statement, the number of records affected, potential errors (for example, unique constraint violations) that were reported, the database connection being used, and so on. Therefore, the SQL trace is a good starting point for understanding executed statements and their potential effect on the overall application and system performance, as well as for identifying potential performance bottlenecks at statement level.

SQL trace information is saved to trace files that you can access on the Diagnosis Files tab of the Administration editor.

Note: Writing SQL trace files can impact database performance significantly. They also consume storage space on the disk. Therefore, it is not recommended that you leave the SQL trace enabled all the time.

SQL Trace Options

You activate the SQL trace in the Administration editor on the Trace Configuration tab.

You can choose from the following trace levels:

<table>
<thead>
<tr>
<th>Trace Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERROR</td>
<td>Information about statements that returned an error is collected.</td>
</tr>
<tr>
<td>ERROR_ROLLBACK</td>
<td>Information about statements that returned an error and/or were rolled back is collected.</td>
</tr>
<tr>
<td>ALL</td>
<td>Information about all statements is collected.</td>
</tr>
<tr>
<td>ALL_WITH_RESULTS</td>
<td>Information about all statements is collected with detailed statistics about results counts. Note that an SQL trace with results can quickly become very large.</td>
</tr>
</tbody>
</table>

The following additional configuration options are available:

- You can specify filters to restrict traced statements to those of particular database or application users and applications, as well as to certain statement types and tables. All statements matching the filter criteria are recorded and saved to the specified trace file.
- You can specify a user-specific name for the trace file.

If you do not enter a user-specific file name, the file name is generated according to the following default pattern:
sqltrace_${HOST}_${PORT}_${COUNT:3}.py, where:

- $HOST is the host name of the service (for example, indexserver)
- $PORT is the port number of the service
- $COUNT:3 is an automatically generated 3-digit number starting with 000 that increments by 1 and serves as a file counter when several files are created.

You can adapt the flush interval if required.

During tracing, the messages of a connection are buffered. As soon as the flush interval number of messages is buffered (or if the connection is closed), those messages are written to the trace file.

Related Links

Monitoring System Performance [page 84]

In addition to the general information about the overall system performance that you can see in the System Monitor and on the Overview tab of the Administration editor, you can monitor more detailed aspects of system performance on the Performance tab.

16.4.2 Performance Trace

The performance trace is a performance tracing tool built into the SAP HANA database. It records performance indicators for individual query processing steps in the database kernel.

Information collected includes the processing time required in a particular step, the data size read and written, network communication, and information specific to the operator or processing-step-specific (for example, number of records used as input and output).

It is recommended that you start performance tracing immediately before running the command(s) that you want to analyze and stop it immediately after they have finished. When you stop tracing, the results are saved to trace files that you can access on the Diagnosis Files tab of the Administration editor. You cannot analyze these files meaningfully in the SAP HANA studio, but instead must use a tool capable of reading the output format (*.tpt). SAP Support has tools for evaluating performance traces.

Note: The performance trace is tool for experts. To interpret the information collected, you require a deep understanding of the system component being analyzed.

Performance Trace Options

You activate the performance trace in the Administration editor on the Trace Configuration tab.

The following configuration options are available:

- You can configure the trace for a single specific database user, a single specific application user, and a single specific application. If you are activating the performance trace as part of an end-to-end trace scenario, you can specify the passport trace level as an additional filter. This means that only requests that are marked with a passport of the specified level are traced.
- You can trace execution plans in addition to the default trace data.
- You can activate the function profiler.

The function profiler is a very fine-grained performance tracing tool based on source code instrumentation. It complements the performance trace by providing even more detailed information about the individual processing steps that are done in the database kernel.
You can specify the file name to which the trace data is automatically saved after the performance trace is stopped.

You can specify how long you want tracing to run.

If a certain scenario is to be traced, ensure that you enter a value greater than the time it takes the scenario to run. If there is no specific scenario to trace but instead general system performance, then enter a reasonable value. After the specified duration, the trace stops automatically.

**Related Links**

*Monitoring System Performance* [page 84]

In addition to the general information about the overall system performance that you can see in the System Monitor and on the *Overview* tab of the Administration editor, you can monitor more detailed aspects of system performance on the *Performance* tab.

### 16.4.3 Kernel Profiler

The kernel profiler is a sampling profiler built into the SAP HANA database. It collects, for example, information about frequent and/or expensive execution paths during query processing.

It can be used to analyze performance issues with systems on which third-party software cannot be installed, or parts of the database that are not accessible by the performance trace.

**Note:**

To be able to use the kernel profile, you must have the SUPPORT role and either the system privilege `RESOURCE_ADMIN` or `TRACE_ADMIN`.

It is recommended that you start kernel profiler tracing immediately before you execute the statements you want to analyze and stop it immediately after they have finished. This avoids the unnecessary recording of irrelevant statements. It is also advisable as this kind of tracing can negatively impact performance. When you stop tracing, the results are saved to trace files that you can access on the *Diagnosis Files* tab of the Administration editor. You cannot analyze these files meaningfully in the SAP HANA studio, but instead must use a tool capable of reading the configured output format, that is KCacheGrind or DOT (default format).

**Note:** The kernel profiler is a performance analysis tool for experts. To interpret the information collected, you require a deep understanding of the source code of the SAP HANA database kernel.

**Kernel Profiler Options**

You activate the kernel profile in the Administration editor on the *Trace Configuration* tab.

When you start tracing, you can specify the following configuration options:

- The amount of time the kernel profiler is to wait between call stack retrievals

  When you activate the kernel profiler, it retrieves the call stacks of relevant threads several times. If a wait time is specified, it must wait the specified time minus the time the previous retrieval took.

- Memory limit that will stop tracing

  The kernel profiler can potentially use a lot a memory. To prevent the SAP HANA database from running out of memory due to profiling, you can specify a memory limit that cannot be exceeded.

When you stop tracing, you can configure the output format.
In addition to the general information about the overall system performance that you can see in the System Monitor and on the Overview tab of the Administration editor, you can monitor more detailed aspects of system performance on the Performance tab.

### 16.4.4 Other Database Traces

Other database traces can be written to obtain detailed information about the actions of the system, including component-specific database traces, traces for end-to-end analysis, and expensive statement traces.

You can activate and configure these in the Administration editor on the Trace Configuration tab. You can access the resulting trace files on the Diagnosis Files tab.

#### Database Traces

The global database, database, and user-specific traces are written for several services of the system (for example, index server and name server) to files `<servicename>.trc`. Some of these traces are always activated by default.

> Note: Not all trace components are visible by default in the Trace Configuration dialog box. To view all additional components, select Show All Components.

The following trace levels are available:

<table>
<thead>
<tr>
<th>Trace Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NONE</td>
<td>Tracing is off. However, information about error situations is still recorded.</td>
</tr>
<tr>
<td>ERROR (default) and FATAL</td>
<td>Information about errors is recorded.</td>
</tr>
<tr>
<td>WARNING</td>
<td>Information about potentially problematic situations is recorded.</td>
</tr>
<tr>
<td>INFO</td>
<td>Information about activity in the database is recorded.</td>
</tr>
<tr>
<td>DEBUG</td>
<td>Detailed information intended for debugging is recorded.</td>
</tr>
</tbody>
</table>

#### End-to-End Traces

End-to-end traces are triggered by applications outside of the SAP HANA database. The default trace levels for the SAP HANA database components are normally sufficient and do not need to be changed.


#### Expensive Statements Trace

The expensive statements trace collects SQL statements whose execution time was above a threshold (1000000 microseconds by default). Expensive statements may reduce the performance of the database. You can also monitor expensive statements on the Performance tab.
In addition to the general information about the overall system performance that you can see in the System Monitor and on the Overview tab of the Administration editor, you can monitor more detailed aspects of system performance on the Performance tab.

## 16.5 Collecting and Downloading Diagnosis Information in SAP HANA Studio

To help SAP Support analyze and diagnose problems with your system, you can collect diagnosis information into a zip file, which you can then download and attach to a support message. It is possible to do this even if a connection to the database is not available.

The following prerequisites apply:

<table>
<thead>
<tr>
<th>Database Connection Status</th>
<th>Prerequisites</th>
</tr>
</thead>
</table>
| Connected                  | • To collect diagnosis information, you must be logged on as a database user with SQL privileges to select system tables and views.  
• To download and delete collected diagnosis information, you must be logged on as the operating system administrator (user <sid>adm) or be able to enter these credentials when prompted. |
| Not connected              | You must be logged on as the operating system administrator (user <sid>adm) or be able to enter these credentials when prompted. |

1. In the Administration editor, choose the Diagnosis Files tab.

   **Note:** If there is no connection to the database, the Administration editor opens in diagnosis mode.

2. Choose Diagnosis Information Collect. The system collects the relevant information by executing the Python script fullSystemInfoDump.py and saves it to a zip file. This may take some time and can be allowed to run in the background.

3. To download the zip file containing the collected diagnosis information, proceed as follows:
   a) Choose Diagnosis Files List. If prompted, enter the user name and password of the operating system administrator (user <sid>adm).

      The Diagnosis Information dialog box opens. The zip file containing the collected diagnosis information is listed together with any other zip files of previously collected information.

   b) Select the relevant zip file and choose Download Collection.

   c) Specify the download location.

4. Optional: Delete any old collections that you no longer need by selecting them and choosing Delete Collections.

Related Links

Diagnosis Information Collected by fullSystemInfoDump.py [page 227]
The Python support script `fullSystemInfoDump.py` collects a range of information from your system.

### 16.6 Collecting Diagnosis Information Using Support Script

Using the Python support script `fullSystemInfoDump.py`, you can collect information from your system, even when it is not possible to access it using SQL. You can then add this information to a support message, for example.

**Note:** You can also complete this task in the SAP HANA studio.

The `fullSystemInfoDump.py` script is part of a server installation and runs from a command line. It is located in the directory `$DIR_INSTANCE/exe/python_support`.

To be able to execute the script, you must be logged on as operating system administrator `<sid>adm user`.

To start the script from its location, enter `python fullSystemInfoDump.py`. If the system can be reached by SQL, the script starts collecting support information. If the system cannot be reached by SQL, the script starts collecting support information but does not export data from system tables. The script creates a zip file containing the collected system information and saves it to the directory `DIR_GLOBAL/sapcontrol/snapshots`. `DIR_GLOBAL` typically points to `/usr/sap/<SID>/SYS/global`.

Note the following:

- The name of the zip file is structured as follows:
  
  `fullsysteminfodump_<SAPLOCALHOSTFULL>_<SAPSYSTEMNAME>_<YYYY>_<MM>_<DD>_<HH>_<MM>_<SS>.zip`.
  The timestamp in the file name is UTC. `SAPLOCALHOSTFULL` and `SAPSYSTEMNAME` are taken from the `sapprofile.ini` file.

- The output directory for the zip file is shown as console output when the script is running, but you can look it up by entering the command `hdbsrvutil -z | grep DIR_GLOBAL=`.

- Usage information can be displayed with the following command: `python fullSystemInfoDump.py -h`.
- Version information can be displayed with the following command: `python fullSystemInfoDump.py --version`.

**Related Links**

*Diagnosis Information Collected by fullSystemInfoDump.py* [page 227]

The Python support script `fullSystemInfoDump.py` collects a range of information from your system.

### 16.7 Diagnosis Information Collected by fullSystemInfoDump.py

The Python support script `fullSystemInfoDump.py` collects a range of information from your system.
Log File

All information about what has been collected is shown as console output and is written to a file named log.txt that is stored in the zip file.

Trace Files

Each of the following trace files is put into a file with the same name as the trace file. For storage reasons, only the trace files from the last 7 days are collected unabridged. From older trace files, only the most recent 10,000 lines are collected.

- $DIR_INSTANCE/<SAPLOCALHOST>/trace/daemon_<SAPLOCALHOST>..<...>.trc
- $DIR_INSTANCE/<SAPLOCALHOST>/trace/indexserver_alert_<SAPLOCALHOST>..<...>.trc
- $DIR_INSTANCE/<SAPLOCALHOST>/trace/indexserver_<SAPLOCALHOST>..<...>.trc
- $DIR_INSTANCE/<SAPLOCALHOST>/trace/nameserver_alert_<SAPLOCALHOST>..<...>.trc
- $DIR_INSTANCE/<SAPLOCALHOST>/trace/nameserver_history.trc
- $DIR_INSTANCE/<SAPLOCALHOST>/trace/nameserver_<SAPLOCALHOST>..<...>.trc
- $DIR_INSTANCE/<SAPLOCALHOST>/trace/preprocessor_alert_<SAPLOCALHOST>..<...>.trc
- $DIR_INSTANCE/<SAPLOCALHOST>/trace/preprocessor_<SAPLOCALHOST>..<...>.trc
- $DIR_INSTANCE/<SAPLOCALHOST>/trace/statisticsserver_alert_<SAPLOCALHOST>..<...>.trc
- $DIR_INSTANCE/<SAPLOCALHOST>/trace/statisticsserver_<SAPLOCALHOST>..<...>.trc

Configuration Files

All configuration files are collected unabridged and stored in a file with the same name as the .ini file:

- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/attributes.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/daemon.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/executor.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/extensions.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/filter.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/global.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/indexserver.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/inifiles.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/localclient.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/mimetypemapping.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/nameserver.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/preprocessor.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/scriptserver.ini
- $DIR_INSTANCE/<SAPLOCALHOST>/exe/config/statisticsserver.ini
$DIR_INSTANCE/<SAPLOCALHOST>/exe/config/validmimetypes.ini

Database System Log Files

The following backup log file is collected unabridged: $DIR_INSTANCE/<SAPLOCALHOST>/trace/backuplog.

Runtime Dump Information

For each index server, a runtime dump containing information about threads, stack contexts, and so on is created and stored in the file indexserver_<SAPLOCALHOST>_<PORT>_runtimedump.trc.

Additional Information Collected If SQL Connection Is Available

All rows of the following system tables and monitoring views are exported into a CSV file with the name of the table:

- SYS.M_INIFILE_CONNECTIONS with CONNECTION_ID > 0
- SYS.M_INIFILE_CONTENTS
- SYS.M_LANDSCAPE_HOST_CONFIGURATION
- SYS.M_SERVICE_STATISTICS
- SYS.M_SERVICE_THREADS
- SYS.M_SYSTEM_OVERVIEW
- SYS.M_TABLE_LOCATIONS
- SYS.M_TABLE_TRANSACTIONS
- _SYS_STATISTICS.STATISTICS_ALERT_INFORMATION
- _SYS_STATISTICS.STATISTICS_ALERT_LAST_CHECK_INFORMATION
- _SYS_STATISTICS.STATISTICS_ALERTS
- _SYS_STATISTICS.STATISTICS_INTERVAL_INFORMATION
- _SYS_STATISTICS.STATISTICS_LASTVALUES
- _SYS_STATISTICS.STATISTICS_STATE
- _SYS_STATISTICS.STATISTICS_VERSION

The first 2,000 rows of all remaining tables in schema _SYS_STATISTICS are exported ordered by column SNAPSHOT_ID.

Additional Information Collected If SQL Connection Is Not Available

All available topology information is exported to a file named topology.txt. It contains information about the host topology in a tree-like structure. The keys are grouped using brackets while the corresponding values are referenced by the symbol =>>. For example:

```bash
[]
[host]
```
[['host', 'ld8521'], ['host', 'ld8521', 'role']
  => worker
[['host', 'ld8521', 'group']
  => default
[['host', 'ld8521', 'nameserver']
  [['host', 'ld8521', 'nameserver', '30501']
    [['host', 'ld8521', 'nameserver', '30501', 'activated_at']
     => 2011-08-09 16:44:02.684
    [['host', 'ld8521', 'nameserver', '30501', 'active']
     => no
    [['host', 'ld8521', 'nameserver', '30501', 'info']
      [['host', 'ld8521', 'nameserver', '30501', 'info', 'cpu_manufacturer']
        => GenuineIntel
      [['host', 'ld8521', 'nameserver', '30501', 'info', 'topology_mem_type']
        => shared
    [['host', 'ld8521', 'nameserver', '30501', 'info', 'sap_retrieval_path_devid']
      => 29
    [['host', 'ld8521', 'nameserver', '30501', 'info', 'build_time']
      => 2011-07-26 17:15:05
    [['host', 'ld8521', 'nameserver', '30501', 'info', 'net_realhostname']
      => -
    [['host', 'ld8521', 'nameserver', '30501', 'info', 'build_branch']
      => orange_COR
    [['host', 'ld8521', 'nameserver', '30501', 'info', 'mem_swap']
      => 34359730176
    [['host', 'ld8521', 'nameserver', '30501', 'info', 'mem_phys']
      => -]
17 SAP Solution Manager for SAP HANA

17.1 Integrating SAP HANA Administration with SAP Solution Manager

You use SAP Solution Manager to manage your business applications throughout their entire lifecycle. As of release 7.1, SP05, SAP Solution Manager supports integration with SAP HANA.

SAP HANA is often used in conjunction with other SAP business applications. For example, an SAP ERP system might call accelerators on SAP HANA to speed up business processes, or a product such as SAP Business Warehouse is deployed on the SAP HANA database. If you are using SAP HANA in such a context, then you must manage your business application in addition to administering the in-memory database. This is best done using an integrated approach.

SAP provides you with the SAP Solution Manager application management platform as part of your maintenance agreement. You can use it to manage your business applications throughout their entire lifecycle. As of release 7.1, SP05, SAP Solution Manager supports integration with SAP HANA. You can optimize your operational processes using this combined approach. One example is root cause analysis. Let’s assume you have detected a problem in an application that is deployed on SAP HANA or calls an SAP HANA accelerator. In this case, you first have to find out whether the problem is caused by the application or by the SAP HANA database. SAP Solution Manager allows you to trace a process across all included components (from the user interface to the database) to locate the source of the problem. Then, detailed analysis speeds up your resolution process.

Other examples of how SAP HANA and SAP Solution Manager can be valuably integrated in the area of system operation are the processes for monitoring and change control. If your business application is still running on a traditional database, even integrated database administration might be relevant.

For more information about these processes, see Using SAP Solution Manager for Administration of SAP HANA. This guide and further information about the integration of SAP HANA and SAP Solution Manager are published on SAP Service Marketplace at https://service.sap.com/solman-hana.

Note: To access content on SAP Service Marketplace, you need an authorized user ID. Do you need to register for an SAP Service Marketplace login? Did you forget your password?
18 SAP HANA HDBSQL

SAP HANA HDBSQL is a command line tool for executing commands on SAP HANA databases.

For example, you can execute SQL statements and database procedures, and query information about the database and database objects. HDBSQL is a component of the SAP HANA software. You can use it to access databases both on your local computer and on remote computers.

You call HDBSQL with the command `hdbsql [options]`. You can execute individual commands interactively or non-interactively. It is also possible to import commands from a file and execute them in the background.

18.1 HDBSQL Commands

You can execute HDBSQL commands in interactive mode to query information about the database and database objects.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>?</code></td>
<td>Displays all HDBSQL commands</td>
</tr>
<tr>
<td><code>h[elp]</code></td>
<td></td>
</tr>
<tr>
<td>`a[utocommit] [ON</td>
<td>OFF]`</td>
</tr>
<tr>
<td>`a[lign] [ON</td>
<td>OFF]`</td>
</tr>
<tr>
<td>`e[scape] [ON</td>
<td>OFF]`</td>
</tr>
<tr>
<td><code>c[onnect]</code></td>
<td>Logs a user onto the database</td>
</tr>
<tr>
<td><code>dc [PATTERN]</code></td>
<td>Lists all table columns that correspond to the specified [PATTERN] and to which the current user has access</td>
</tr>
</tbody>
</table>

[PATTERN] can be specified as follows: [SCHEMA.] [OBJECT_NAME]. The following placeholders are possible:

- For one character: `_`
- For any number of characters: `%`

If a pattern is not specified, the system returns information about all table columns to which the current user has access.

This command returns the following information:

- Column name

Note: Instead of an HDBSQL command, you can also enter an SQL statement or a database procedure. The statement or procedure must be in quotation marks.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| `\de [PATTERN]` | Lists all the indexes of database objects that correspond to the specified [PATTERN]. [PATTERN] can be specified as follows: [SCHEMA.] [OBJECT_NAME]. The following placeholders are possible:  
• For one character: `_`  
• For any number of characters: `%`  
If a pattern is not specified, the system returns information about all indexes of database objects to which the current user has access. This command returns the following information:  
• Index name  
• Columns contained in index  
• Position of column in index  
• Specifies whether index is UNIQUE  
• Sort sequence |
| `\ds [NAME]` | Lists all schemas that correspond to the specified [NAME]. [NAME] can be specified as follows: [SCHEMA.] [OBJECT_NAME]. The following placeholders are possible:  
• For one character: `_`  
• For any number of characters: `%`  
If a pattern is not specified, the system returns information about all schemas to which the current user has access. |
| `\dp [PATTERN]` | Lists all database procedures that correspond to the specified [PATTERN]. [PATTERN] can be specified as follows: [SCHEMA.] [OBJECT_NAME]. The following placeholders are possible:  
• For one character: `_`  
• For any number of characters: `%`  
If a pattern is not specified, the system returns information about all database procedures to which the current user has access. This command returns the following information:  
• Schema name  
• Name of the database procedure  
• Package to which database procedure is assigned |
<p>| <code>\di[sconnect]</code> | Logs the user off from the database |</p>
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| \dt [PATTERN] | Lists all tables that correspond to the specified [PATTERN]  
[PATTERN] can be specified as follows: [SCHEMA.] [OBJECT_NAME]. The following placeholders are possible:  
- For one character: _  
- For any number of characters: %  
If a pattern is not specified, the system returns information about all tables to which the current user has access.  
This command returns the following information:  
- Schema name  
- Table name  
- Table type                                                                                                                                                                                                 |
| \du [NAME]  | Lists all database users that correspond to the specified [NAME]  
[NAME] can be specified as follows: [SCHEMA.] [OBJECT_NAME]. The following placeholders are possible:  
- For one character: _  
- For any number of characters: %  
If a name is not specified, the system returns information about all database users to which the current user has access.  
This command returns the following information:  
- Name of the database user  
- User properties                                                                                                                                                                                                 |
| \dv [PATTERN] | Lists all views that correspond to the specified [PATTERN]  
[PATTERN] can be specified as follows: [SCHEMA.] [OBJECT_NAME]. The following placeholders are possible:  
- For one character: _  
- For any number of characters: %  
If a pattern is not specified, the system returns information about all views to which the current user has access.  
This command returns the following information:  
- Schema name  
- View name  
- View types                                                                                                                                                                                                 |
| \e[dit]<file>  | Writes the command buffer to the <file> where you can edit it with an editor                                                                                                                                                                                                                                                                 |
### 18.2 HDBSQL Command Line Options

HDBSQL command line options modify the operation of HDBSQL commands.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-i &lt;instance_id&gt;</td>
<td>Instance ID of the database</td>
</tr>
<tr>
<td>-e</td>
<td>Encrypted data transmission</td>
</tr>
<tr>
<td>-n &lt;database_computer&gt;[:&lt;port&gt;]</td>
<td>Name of the computer on which the database is installed and port number</td>
</tr>
<tr>
<td>-r</td>
<td>Enforces execution of SQL statements as statements rather than as prepared statements</td>
</tr>
<tr>
<td>-u &lt;database_user&gt;</td>
<td>User name for logging on to the database</td>
</tr>
<tr>
<td>-p &lt;database_user_password&gt;</td>
<td>Password for logging on to the database</td>
</tr>
<tr>
<td>-U &lt;user_store_key&gt;</td>
<td>Uses credentials from user store</td>
</tr>
<tr>
<td>-S &lt;sql_mode&gt;</td>
<td>SQL mode, either &quot;INTERNAL&quot; or &quot;SAPR3&quot;</td>
</tr>
<tr>
<td>-z</td>
<td>Switches AUTOCOMMIT mode off</td>
</tr>
<tr>
<td>-r</td>
<td>Suppresses usage of prepared statements</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>-saml-assertion &lt;file&gt;</td>
<td>Uses a file to provide a SAML assertion</td>
</tr>
</tbody>
</table>

Table 33: Input and Output

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-c &lt;separator&gt;</td>
<td>When importing commands from a file, HDBSQL uses the &lt;separator&gt; to separate the individual commands. The default value is ;.</td>
</tr>
<tr>
<td>-I &lt;file&gt;</td>
<td>Imports commands from the batch file &lt;file&gt;</td>
</tr>
<tr>
<td>-m</td>
<td>Activates multiple line mode for entering HDBSQL commands</td>
</tr>
<tr>
<td>-o &lt;file&gt;</td>
<td>Writes the results to the &lt;file&gt;</td>
</tr>
<tr>
<td>-x</td>
<td>Suppresses additional output such as the number of selected rows in a result set</td>
</tr>
<tr>
<td>-resultencoding &lt;encoding&gt;</td>
<td>Forces output encoding, one of &quot;UTF8&quot;, &quot;LATIN1&quot; or &quot;AUTO&quot; (AUTO is default) for result data</td>
</tr>
</tbody>
</table>

Table 34: Formatting Output

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-A</td>
<td>Returns the result set in an aligned format</td>
</tr>
<tr>
<td>-a</td>
<td>Suppresses the output of the column names in the result set</td>
</tr>
<tr>
<td>-C</td>
<td>Suppresses escape output format</td>
</tr>
<tr>
<td>-b &lt;maximum_length&gt;</td>
<td>Defines the maximum number of characters for output of LOB values. Default value is 10 characters.</td>
</tr>
<tr>
<td>-f</td>
<td>HDBSQL returns all SQL statements before sending them to the database instance</td>
</tr>
<tr>
<td>-F &lt;separator&gt;</td>
<td>Specifies which string HDBSQL uses as a separator between the individual columns of the result set Default value is .</td>
</tr>
<tr>
<td>-g &lt;null_value&gt;</td>
<td>Specifies the character for null values in the result set Default value is ?</td>
</tr>
<tr>
<td>-p &lt;prefix&gt;</td>
<td>Specifies which string is to be output before each row of the result set Default value is .</td>
</tr>
<tr>
<td>-P &lt;suffix&gt;</td>
<td>Specifies which string is to be output after each row of the result set</td>
</tr>
</tbody>
</table>
### Table 35: Other

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-Q</td>
<td>Outputs each column of the result set in a new row</td>
</tr>
<tr>
<td>-j</td>
<td>Switches the page by page scroll output off</td>
</tr>
</tbody>
</table>

### Table 36: SSL Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-sslprovider</td>
<td>SSL provider [sapcrypto</td>
</tr>
<tr>
<td>-sslkeystore</td>
<td>SSL key store name</td>
</tr>
<tr>
<td>-ssltruststore</td>
<td>SSL trust store name</td>
</tr>
<tr>
<td>-ssltrustcert</td>
<td>Skip certificate validation</td>
</tr>
<tr>
<td>-sslhostnameincert</td>
<td>Hostname used for certificate validation</td>
</tr>
<tr>
<td>-sslcreatecert</td>
<td>Create self-signed certificate</td>
</tr>
</tbody>
</table>

### 18.3 Logging On to a Database

To use HDBSQL interactively and to execute some commands, you must log on to the database as a database user.

**Note:** The user logging on must be a database user. If you do not specify a username and password of a database user, logon is attempted using Kerberos authentication.

<table>
<thead>
<tr>
<th>Logon Option</th>
<th>How</th>
</tr>
</thead>
<tbody>
<tr>
<td>One-step logon with username and password</td>
<td>Enter the following command:</td>
</tr>
</tbody>
</table>
Logon Option | How
---|---
hdbsql [<options>] -n <database_host> -i <instance_id> -u <database_user> - p <database_user_password>

Two-step logon with username and password

1. Start HDBSQL by entering the following command:
   ```
   hdbsql [<options>]
   ```
2. Log on to the database by entering the following command:
   ```
   \c [<options>] -n <database_host> -i <instance_id> -u <database_user>, -p <database_user_password>
   ```

**Note:** You can log on with user credentials for the user store with `-U <user_key>`.

For one-step logon to the database on the PARMA computer with instance ID 01 as database user MONA with the password RED, the command is as follows:

```
hdbsql -n PARMA -i 1 -u MONA -p RED
```

### 18.4 Executing Commands

You can execute HDBSQL commands in interactive and non-interactive mode.

To execute some commands, you must be logged on to the database.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Steps</th>
</tr>
</thead>
</table>
| Interactive (session) mode | 1. Call HDBSQL by entering the following command:
   ```
   hdbsql [options]
   ```

   2. Enter the command and press *Enter*.

   HDBSQL executes the command.

   **Note:**
   
   To execute an SQL statement or a database procedure as a command, place the statement or procedure in quotation marks.

   3. Exit HDBSQL by entering the following command:
   ```
   exit | quit | \q
   ```
Non-interactive (command) mode

Call HDBSQL and enter the command as follows:

```
hdbsql [options] <command>
```

HDBSQL executes the command and then exits.

**Note:** To execute an SQL statement or a database procedure as a command, place the statement or procedure in quotation marks.

In addition to executing commands individually, you can execute multiple commands from a batch file. HDBSQL imports the commands from the specified file and processes them in the background.

You can specify the separator used in the batch file to separate individual commands using the `-c <separator>` command line option. The default value is `;`.

**Note:** If you execute from a batch file, AUTOCOMMIT mode is activated by default. If you deactivate it, the batch file must contain an explicit COMMIT statement to ensure that HDBSQL executes the SQL statements immediately after the batch file has been imported.

**Example 1**

Individual command execution to display general information about the database in command mode, with simultaneous database logon:

```
hdbsql -n localhost -i 1 -u USER1 -p Password123 \
```

Result:

```
host: wdfd00245293a:30015
database: ORG
user: USER1
kernel version: 1.00.38.368649
SQLDBC version: libSQLDBCHDB 1.00.38.368649 Build 0000000-0120
autocommit: ON
```

**Example 2**

Individual execution of SELECT statement in command mode, with simultaneous database logon:

```
hdbsql -n localhost -i 1 -u USER1 -p Password123 "select
CNO,TITLE,FIRSTNAME,NAME,ZIP from HOTEL.CUSTOMER"
```

Result:

```
CNO | TITLE | FIRSTNAME | NAME  | ZIP
--------+--------+-----------+-------+------
3000 | Mrs   | Jenny     | Porter | 10580
3100 | Mr    | Peter     | Brown  | 48226
3200 | Company | ?        | Datasoft | 90018
```
Example 3

Execution of multiple commands imported from a batch file in command mode:

```
hdbsql [<options>] -I <file>
```

The file contains the following commands for execution:

```
CREATE TABLE city
(zip NCHAR (5) PRIMARY KEY,
name NCHAR(20),
state NCHAR(2));
CREATE TABLE customer
(cno INTEGER PRIMARY KEY,
title NCHAR (7),
firstname NCHAR (10),
name NCHAR (10),
zip NCHAR (5),
address NCHAR (25));
\dt customer;
COMMIT
```

18.5 Executing Long Commands in Multiple Line Mode

Multiple line mode enables you to enter long commands on several lines, for example, a long SQL command. HDBSQL stores multiple line commands in an internal command buffer.

To execute some commands, you must be logged on to the database.

1. **Activate multiple line mode with one of the following commands:**
   - Call option: `hdbsql [<options>] -m`
   - HDBSQL command: `\mu ON`

2. **Enter the command.**
   - To start a new line, press `Enter`

3. **Execute the command in one of the following ways:**
   - Close the last line of the command by entering a semicolon and press `Enter`
Enter the command \g.

1. Log on to the SAP HANA database as user MONA with the password RED by entering the following:
   
   ```
   hdbsql -n localhost -i 1 -u MONA,RED
   ```

2. Activate multiple line mode:
   
   ```
   \mu ON
   ```

3. Enter a multiple line SQL statement:
   
   ```
   SELECT ROUND(SUM("M")/1024/1024/1024,2) as "Peak Used Memory GB"
   FROM
   (SELECT SUM(CODE_SIZE+SHARED_MEMORY_ALLOCATED_SIZE) AS "M" FROM
   SYS.M_SERVICE_MEMORY UNION SELECT
   SUM(INCLUSIVE_PEAK_ALLOCATION_SIZE) AS
   "M" FROM M_HEAP_MEMORY_RESET WHERE DEPTH = 0)
   ```

4. Execute the SQL statement by entering the command \g.

18.6 Editing Long Commands in an External File

If you have entered a long command in HDBSQL in multiple line mode, you can change it later by editing the command buffer in an external file and then re-executing it.

- **Note**: You must have already executed the command.

1. To export the contents of the command buffer to an external file, enter the following command:
   
   ```
   \o [<file>]
   ```

   - **Note**: You must enter the complete file path and <file> name. If you do not specify a file, HDBSQL generates a temporary file.

   The system opens the file in an editor. To determine which editor is used, HDBSQL evaluates the environment variables `<HDBSQL_EDITOR>`, `<EDITOR>`, and `<VISUAL>` in succession. If you have not set any of these environment variables, vi is used on Linux and Unix. For more information about setting environment variables, see your operating system documentation.

2. Make the desired changes to the file.

3. Save the file in the editor then close file and editor.

You have changed the contents of the command buffer and can now execute the changed command with the command \g.

18.7 Redirecting Results to a File

You can redirect the result of one or more HDBSQL commands to a file.
To redirect results to a file, you must be logged on to the database.

1. Enter the following command:
   \o <file>
   
   **Note:** You must enter the full path of the file.

2. Enter the command the result of which is to be redirected to the file.
   To enter multiple commands in succession, press enter after each command.

3. To stop redirection to a file, enter \o.

You want to export a list of all schemas, as well as all entries in the table HOTEL.CUSTOMER to an external file.

1. Log on to the SAP HANA database as user MONA with the password RED by executing the command:
   ```
   hdbsql -n localhost -i 1 -u MONA,RED
   ```

2. Create the file c:\tmp\redirected.txt to which HDBSQL is to redirect the result by executing the command:
   ```
   \o c:\tmp\redirected.txt
   ```

3. Request information about all schemas by executing the command:
   ```
   \ds
   ```

4. Select all rows in the table HOTEL.CUSTOMER by executing the command:
   ```
   SELECT * from HOTEL.CUSTOMER
   ```

5. Stop redirection to the file by entering \o.

<table>
<thead>
<tr>
<th>Schema</th>
<th>Owner name</th>
</tr>
</thead>
<tbody>
<tr>
<td>------</td>
<td>---------------</td>
</tr>
<tr>
<td>MDX_TE</td>
<td>SYSTEM</td>
</tr>
<tr>
<td>SECURI</td>
<td>SECURITY1</td>
</tr>
<tr>
<td>SOP.PL</td>
<td>SYSTEM</td>
</tr>
<tr>
<td>SYS</td>
<td>SYS</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>SYSTEM</td>
</tr>
<tr>
<td>_SYS_B</td>
<td>_SYS_REPO</td>
</tr>
<tr>
<td>_SYS_B</td>
<td>_SYS_REPO</td>
</tr>
<tr>
<td>_SYS_R</td>
<td>_SYS_REPO</td>
</tr>
<tr>
<td>_SYS_S</td>
<td>_SYS_STATISTICS</td>
</tr>
<tr>
<td>CNO</td>
<td>TITLE</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td>3200</td>
<td>Company</td>
</tr>
<tr>
<td>3400</td>
<td>Mrs</td>
</tr>
<tr>
<td>3500</td>
<td>Mr</td>
</tr>
<tr>
<td>3600</td>
<td>Mrs</td>
</tr>
<tr>
<td>3700</td>
<td>Mr</td>
</tr>
<tr>
<td>3900</td>
<td>Mr</td>
</tr>
<tr>
<td>4000</td>
<td>Mr</td>
</tr>
<tr>
<td>4400</td>
<td>Mr</td>
</tr>
</tbody>
</table>