Configuration and Users Guide

Replication Management Agent (RMA) 15.7.1 SP200

DOCUMENT ID: DC02002-01-1571200-01
LAST REVISED: March 2014

Contents

Conventions

These style and syntax conventions are used in SAP® documentation.

Style conventions

Key	Definition
`monospaced(fixed-width)`	- SQL and program code - Commands to be entered exactly as shown - File names - Directory names
`italic monospaced`	In SQL or program code snippets, placeholders for user-specified values (see example below).
italic	- File and variable names - Cross-references to other topics or documents - In text, placeholders for user-specified values (see example below) - Glossary terms in text
bold san serif	- Command, function, stored procedure, utility, class, and method names - Glossary entries (in the Glossary) - Menu option paths - In numbered task or procedure steps, user-interface (UI) elements that you click, such as buttons, check boxes, icons, and so on

If necessary, an explanation for a placeholder (system- or setup-specific values) follows in text. For example:

Run:

```
installation directory\start.bat
```

where *installation directory* is where the application is installed.

Syntax conventions

Key	Definition
{ }	Curly braces indicate that you must choose at least one of the enclosed options. Do not type the braces when you enter the command.
[ ]	Brackets mean that choosing one or more of the enclosed options is optional. Do not type the brackets when you enter the command.
( )	Parentheses are to be typed as part of the command.
|	The vertical bar means you can select only one of the options shown.
,	The comma means you can choose as many of the options shown as you like, separating your choices with commas that you type as part of the command.
...	An ellipsis (three dots) means you may repeat the last unit as many times as you need. Do not include ellipses in the command.

Case-sensitivity

- All command syntax and command examples are shown in lowercase. However, replication command names are not case-sensitive. For example, **RA_CONFIG**, **Ra_Config**, and **ra_config** are equivalent.
- Names of configuration parameters are case-sensitive. For example, **Scan_Sleep_Max** is not the same as **scan_sleep_max**, and the former would be interpreted as an invalid parameter name.
- Database object names are not case-sensitive in replication commands. However, to use a mixed-case object name in a replication command (to match a mixed-case object name in the primary database), delimit the object name with quote characters. For example: **pdb_get_tables** "*TableName*"
- Identifiers and character data may be case-sensitive, depending on the sort order that is in effect.
  - If you are using a case-sensitive sort order, such as “binary,” you must enter identifiers and character data with the correct combination of uppercase and lowercase letters.
  - If you are using a sort order that is not case-sensitive, such as “nocase,” you can enter identifiers and character data with any combination of uppercase or lowercase letters.

Terminology

SAP® Replication Server® works with various components to enable replication between supported database such as, SAP® Adaptive Server® Enterprise (SAP® ASE), SAP HANA® database, SAP® IQ, Oracle, IBM DB2 UDB, and Microsoft SQL Server. SAP Replication Server uses SAP ASE for its Replication Server System Database (RSSD) or it uses SAP® SQL Anywhere® for its embedded Replication Server System Database (ERSSD).

Replication Agent™ is a generic term used to describe the Replication Agents for SAP ASE, SAP HANA database, Oracle, IBM DB2 UDB, and Microsoft SQL Server. The specific names are:

- RepAgent – Replication Agent thread for SAP ASE
- Replication Agent for Oracle
- Replication Agent for Microsoft SQL Server
- Replication Agent for UDB – for IBM DB2 on Linux, Unix, and Windows
- Replication Agent for DB2 for z/OS

Introduction to Replication Management Agent

Replication Management Agent (RMA) is a distributed management agent that you can use to set up and manage replication from any supported databases to an SAP HANA database.

RMA supports automated setup and configuration, monitoring, and administration of an SAP Business Suite or non-Business Suite system.

Using RMA, you can set up replication for these primary databases:

- IBM UDB DB2
- Microsoft SQL Server
- Oracle

RMA Replication Architecture

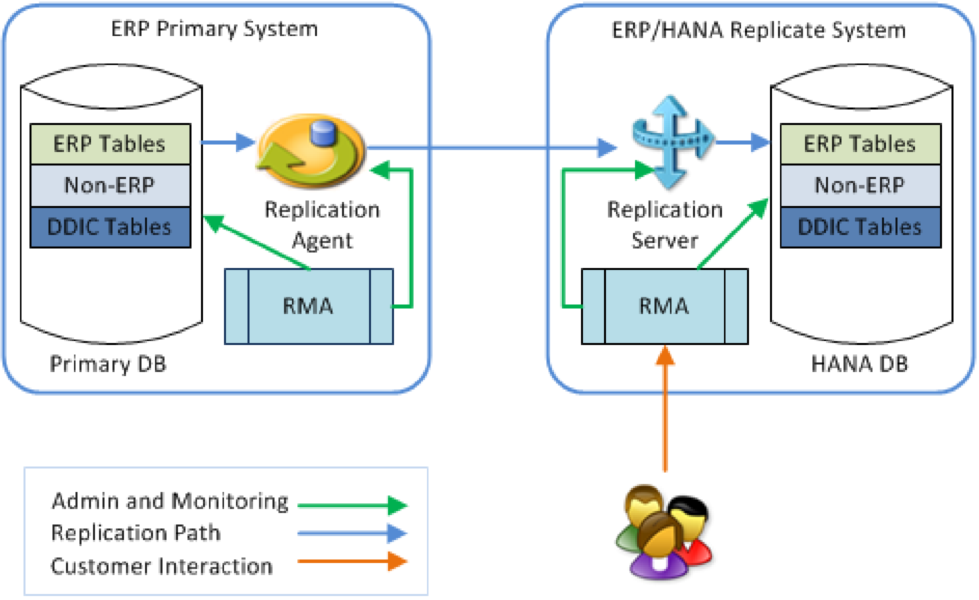
RMA replication architecture diagram illustrates a very basic replication environment, which consists of the primary and the replicate sites.

Each site, whether primary or replicate, contains a database and optionally a portion of the replication software. Each site contains replication software and an RMA instance that provides local access to the host system, when needed.

Replication is set up to support replication in one direction between the primary and replicate. In a typical replication scenario, the SAP application connects to and updates data on the primary site. The replication software captures transactions, and replicates to the replicate site.

During the initial setup and configuration, you connect to the RMA client on the primary or replicate site, and issue commands or follow instructions to define the environment and set up replication. After the replication environment is set up, you can also connect to an RMA client on either site to monitor and administer the replication environment.

Your replication software may be installed on multiple servers. It is often advantageous to install the Replication Agent on the same host as the primary database, and the SAP Replication Server on the same host as the replicate database. This spreads processing across multiple servers and allows database communication to be near the database that is being communicated with, allowing local client library installations to be used. However, you can install the replication software on the same host or totally separate hosts, as all communication is over TCP/IP. The replication environment diagram is just one example.


ERP Primary System
ERP Tables
Non-ERP
DDIC Tables
Primary DB
Replication Agent
RMA
ERP/HANA Replicate System
Replication Server
RMA
ERP Tables
Non-ERP
DDIC Tables
HANA DB
Admin and Monitoring
Replication Path
Customer Interaction

Before You Begin

Before using RMA, install and configure all the replication components (SAP Replication Server, Replication Agent, primary and replicate databases).

See the *Quick Start Guide for SAP HANA Database*.

Primary and Replicate Databases Prerequisites

Ensure that the primary and replicate databases meet the environment requirements for using RMA.

UDB DB2

These items are required for your UDB DB2 database environment:

- User with DBADM authority – the Replication Agent requires the use of a UDB login that has DBADM authority. This is required to access the UDB log read API feature.

  This is an example of a **grant** command that you can use to grant dbadm permission to a UDB user:

  ```
  grant dbadm on database to user login_name
  ```

  This is an example of a **select** command that you can use to list all users who currently have dbadm authority:

  ```
  select grantee from SYSCAT.DBAUTH where DBADMAUTH =‘Y‘
  ```
- User to identify as maintenance user – the Replication Agent requires a UDB login to be identified as the maintenance user for the database. Maintenance user activity is excluded or filtered from replication (the assumption is that maintenance user transactions generate activity from replication, and this activity must be excluded from further replication to avoid circular redundancy). This user requires no special permissions in this environment, but must be a user different than the DBADM user, so that the DBADM user activity can be replicated.
- Enable log archiving in the database, so the UDF log read API can function – archiving is enabled when one of the database configurations— "logarchmeth1" or "logarchmeth2"— is set to “LOGRETAIN”, or points to a disk file location or TSM management class. See the IBM documentation for more information about UDB archiving.
- Create a user temporary table space, or verify that one exist – some UDB stored procedures or functions that the Replication Agent executes require the user to have access to a temporary table space to store temporary results. The table space must be at least 8K in size.

SAP HANA Database

These items are required for your HANA database environment:

- Dedicated maintenance user – the replication process requires a dedicated HANA login that is used to maintain or apply all replication-related commands. This user must have **insert**/**update**/**delete** and truncate authority on any schema and table that is to be replicated.

  See the HANA documentation at *http://help.sap.com/hana/html/sql_grant.html* for more information about grants and permissions.
- User with DBA authority – the RMA supports the identification of a HANA login that has DBA authority. Identify the maintenance user also as the administrative user.

Managing the Replication Environment with RMA

Use the RMA to manage and monitor your replication environment.

Starting the RMA on UNIX

Run the RMA on a UNIX platform.

Prerequisites

Set the environment variables:

1. Log in to the host using the same operating system user that was used to install the RMA and SAP Replication Server.
2. Set the necessary environment variables by running the appropriate command.
   - Bourne shell:

     ```
     . SYBASE.sh
     ```
   - C shell:

     ```
     source SYBASE.csh
     ```

Note: Set the environment variables only once.

Task

1. Change to the `bin` directory of the RMA installation:

   ```
   cd /sybase/SAP_SID_REP/RMA-15_5/bin
   ```
2. Execute the appropriate command for your shell.

   These examples direct the output to the file `rma.out`. If this output file already exists, you may need to use additional shell operators to append to or truncate the file.

   - Bourne shell (sh) or Bash:

     -
       ```
       nohup ./RunContainer.sh 2>&1 > rma.out &
       ```
   - For an instance, C shell:

     ```
     nohup ./RunContainer.sh >& rma.out &
     ```

Starting the RMA on Windows

Run the RMA on both the primary and replicate sites on Windows.

Prerequisites

Verify that you have the administrator privilege, which allows you to create and delete Windows services for SAP Replication Server and Replication Agent.

Task

1. Log in to the host using the same operating system user that was used to install the RMA and SAP Replication Server.
2. Open a command window and run as administrator:
   a) Click the Windows start button.
   b) Enter `cmd` in the Search Programs and Files window.
   c) Right-click **cmd.exe**, and choose **Run as administrator**.
3. From the command window, navigate to the directory where the RMA is installed.
4. Run `SYBASE.bat` to set the necessary environment variables.
5. Change to the `bin` directory of the RMA installation:

   ```
   cd /sybase/SAP_SID_REP/RMA-15_5/bin
   ```
6. Enter `RunContainer.cmd` to start the RMA.

Verifying the RMA

(UNIX only) Verify that the RMA process is running.

In addition to logging in to RMA with a client to determine availability, **grep** the Java process to verify that it is running on UNIX boxes within your installation directory.

```
ps -ef | grep D01_REP | grep
com.sybase.rs.container.runtime.ContainerRuntime
```

Shutting Down the RMA

Shut down from a client session attached to RMA.
Execute:

```
shutdown
```

Connecting to the SAP Replication Management Agent Designer

Connect to the SAP Replication Management Agent Designer to use RMA.

Prerequisites

If you are connecting to the SAP Replication Management Agent Designer on the UNIX platform, verify that you have xWindows support on your local machine or UNIX systems.

Note: If you do not have xWindows support, the client stops without error. Currently, OpenText Exceed is the only supported xWindows software for Windows.

Task

1. Set the environment variables.
   - UNIX:

     Source `SYBASE.sh` or `SYBASE.csh`
   - Windows:

     Run `SYBASE.bat`
2. Change to the `bin` directory of the RMA installation.
   - UNIX

     ```
     cd /sybase/SAP_SID_REP/RMA-15_5/bin
     ```
   - Windows:

     ```
     cd C:\sybase\SAP_SID_REP\RMA-15_5\bin
     ```
3. Start the SAP Replication Management Agent Designer.
   - UNIX:

     Run **AdminCC.sh**
   - Windows:

     Run **AdminCC.cmd**

Setting Up the Replication Environment

Use the RMA to set up the replication environment, configuring each host to support replication.

An SAP Replication Server and Replication Agent instances are created and configured on the primary host server. One primary database is configured to support replication. The server supports multiple databases, but only one may be replicated by a single RMA instance.

RMA Security

The RMA does not store its own security credentials, but enforces authentication by pass-through authentication to the primary or replicate SAP Replication Server.

For credentials that are required to connect to the databases, the credentials are identified at setup time, but RMA does not store those credentials.

RMA Error Handling

There are some situations under which a command may fail because of conditions in the environment. RMA handle errors

For example, the **setup** command may fail if the host computer does not have enough disk space to create a replication device. When an error occurs in the RMA, the user can make the necessary environment changes, then re-run the failed command. The RMA starts executing where the last failure occurred. Errors, information, and trace messages are written to the RMA log file, which is located under the RMA instance directory structure, for example, `/sybase/D01_REP/RMA/instance/AgentContainer/logs`.

Setting the Server Session Credentials

Set the login ID and password to access the RMA server for all activities.

1. In the SAP Replication Management Agent Designer window, expand the system name in the Agent Servers pane.
2. Select **AgentContainer**.

   The current properties of the selected container appeared in the right pane.
3. In the Properties tab under the Session Credentials section, enter the login ID and password.

Deploying a Replication Model

To deploy a replication model, provide property values for each replication component.

1. In the SAP Replication Management Agent Designer window, right-click the highlighted **AgentContainer** and select **Deploy a Model**.

   The Replication Model Deployment wizard appears in a separate window.
2. (Optional) If you have a previously saved replication model properties file, click **Load**, browse through your local directory, select the file, and click **Open**.
3. Select a primary database to use for replication from the list, and specify its required configuration information.

   (Optional) Click **Save** to store the information for later use.
4. Click **Next**.

5. Specify all the required configuration information for the replicate database.

   (Optional) Click **Save** to store the information for later use.
6. Click **Next**.
7. Specify all the required configuration information for the Replication Agent.

   (Optional) Click **Save** to store the information for later use.
8. Click **Next**.
9. Specify all the required configuration information for the SAP Replication Server.

   (Optional) Click **Save** to store the information for later use.
10. Click **Next**.
11. Review the Replication Model Summary.

    (Optional) Click **Save** to store the summary of replication model.
12. Click **Deploy**.

If the deployment does not complete successfully, you see an error message in the deployment progress window. The information in this window may help you fix the problem. The Agent Container log also has useful information regarding the errors.

After a successful deployment, the replication model is visible in the Agent Servers pane. To see it, expand the **AgentContainer**.

Primary Database Configuration Properties

When deploying a replication model using RMA, some configuration properties are required for the primary database .

Table 1. UDB DB2 Configuration Properties

Property Name	Description
UDB Host	The host machine where UDB is running.
UDB Port	The TCP/IP port for UDB.
UDB Home Directory	The home directory for the UDB installation user (pointed to by environment variable $HOME when logged on as the same user who installed the UDB.)
Instance Name	The name of the UDB instance.

Property Name	Description
Alias Name	(Optional) The alias for the UDB, typically required only when RMA is installed on the remote host from the primary host. The primary database remote client allows an alias to be used to reference the remote primary database server. The value can be empty or the same name as the udb_name, if RMA is running on the same host as the UDB database.
Archive Directory	The the directory where archive files are located. This property must exist, but can be set to an empty or valid temp directory.
Remove Archive Files	
UDB Admin User	The administrator user name for UDB.
UDB Admin Password	The administrator user password for UDB.
UDB Maintenance User	The maintenance user name for UDB.
UDB Maintenance Password	The maintenance user password for UDB.

Table 2. Oracle Configuration Properties

Property Name	Description
Oracle Host	The host machine where Oracle is running.
Oracle Port	The TCP/IP port for Oracle.
Oracle Home Directory	The home directory for the Oracle installation user (pointed to by environment variable $HOME when logged on as the same user who installed the Oracle.)
Instance Name	The Oracle instance.
Alias Name	(Optional) The alias for the Oracle, typically required only when RMA is installed on the remote host from the primary host. The primary database remote client allows an alias to be used to reference the remote primary database server. The value can be empty or the same name as the oracle_name, if RMA is running on the same host as the Oracle database.

Property Name	Description
Archive Directory	The the directory where archive files are located. This property must exist, but can be set to an empty or valid temp directory.
Oracle Admin User	The administrator user name for Oracle.
Oracle Admin Password	The administrator user password for Oracle.
Oracle Maintenance User	The maintenance user name for Oracle
Oracle Maintenance Password	The password of the maintenance user for Oracle

Table 3. Microsoft SQL Server Configuration Properties

Property Name	Description
MSSQL Host	The host machine where MSSQL is running.
MSSQL Port	The TCP/IP port for Oracle.
MSSQL Home Directory	The home directory for the MSSQL installation user (pointed to by environment variable $HOME when logged on as the same user who installed the MSSQL.)
Server Name	The server name for MSSQL.
Database Name	the database name for MSSQL.
Alias Name	(Optional) The alias for the MSSQL, typically required only when RMA is installed on the remote host from the primary host. The primary database remote client allows an alias to be used to reference the remote primary database server. The value can be empty or the same name as the mssql_name, if RMA is running on the same host as the MSSQL database.
MSSQL Admin User	The administrator user name for MSSQL.
MSSQL Admin Password	The adminisrator user password for MSSQL.
MSSQL Maintenance User	The maintenance user name for MSSQL.
MSSQL Maintenance Password	The maintenance user password for MSSQL.

Replicate Database Configuration Properties

When deploying a replication model using RMA, some configuration properties are required for the SAP HANA database.

Property Name	Description
HANA Host	The host machine on which HANA is running.
HANA Port	The TCP/IP port on which the HANA instance is listening. The default is typically 3xx15, where ‘xx’ is the HANA instance number.
HANA Home Directory	The home directory for the HANA installation. Currently not used.
HANA Instance Name	The logical name for the HANA database. Can be any value and is not validated, but is used to uniquely identify the HANA instance as if this name represents the HANA database *name*.
Replication Delay (minutes)	
Alias Name	Not currently used, but you must provide an empty value.
HANA Admin User Name	The administrator user name for HANA.
HANA Admin Password	The administrator user password for HANA.
HANA Maintenenace User Name	The maintenance user name for HANA.
HANA Maintenance Password	The maintenance user password for HANA.

Replication Server Configuration Properties

When deploying replication model using RMA, some configuration properties are required for the SAP Replication Server.

Property Name	Description
Logical Host Name	The host name for the RMA and SAP Replication Server.
Replication Server Host	The SAP Replication Server instance.
RMA Port	The RMA TDS port. Default value: 7002.
Replication Server Port	The TCP/IP port on which the data applier listens.
RSSD Port	The TCP/IP port the data applier assigns to its system database.
Sybase Directory	The installation directory in which the RMA and SAP Replication Server components are installed.

Property Name	Description
Device Buffer Directory	The directory where files are created that hold or buffer data to be replicated by the data applier.
Device Buffer Size (MB)	The initial allocation, in megabytes, that a static file acquires in the directory specified by the **device_buffer_dir**. This is the minimum buffer that is always available to the data applier. 60MB is the recommended minimum.

Replication Agent Configuration Properties

When deploying a replication model using RMA, some configuration properties are required for the Replication Agent.

Property Name	Description
Logical Host Name	The name that is given to the data collector or Replication Agent instance.
Replication Agent Host	The host name for the RMA and Replication Agent.
RMA Port	The RMA TDS port. Default value: 7002.
Replication Agent Port	The TDS port on which the Replication Agent instance listens. The Replication Agent also uses this port number +1; the two sequential ports are used starting from, and including, this value.
Sybase Directory	The installation directory in which the RMA and Replication Agent components are installed.

Undeploying a Replication Model

Undeploy the replication model.

1. In the SAP Replication Management Agent Designer, expand **AgentContainer**.
2. Right-click the **RMA** node, and select **Undeploy RMA**.
   If your credentials are loaded, an undeploy confirmation dialog appears, otherwise a dialog prompting for credentials appears.
3. If you are prompted for credentials, either:
   - Enter the admin and maintenance user credentials for each component manually, or,
   - Click **Load** to use a previously saved set of credentials.
4. Click **OK**.

Listing Tables in the Replication Environment

View the list of tables in the replication environment.

Prerequisites

Deploy a replication model.

Task

1. In the SAP Replication Management Agent Designer, expand **AgentContainer > RMA**.
2. Right-click **(REPPATH) reppath** and select **Setup Table Replication**.
   The tables in your replication environment are listed on the right side of the SAP Replication Management Agent Designer window.

See also

- *Deploying a Replication Model* on page 12

Configuring Tables for Replication

Specify the tables that you want to replicate and materialize.

Prerequisites

Deploy a replication model.

Task

1. In the SAP Replication Management Agent Designer, expand **AgentContainer > RMA**.
2. Right-click **(REPPATH) reppath**, and select **Setup Table Replication**.
3. Click one or more tables to replicate.

   You can filter the table list using the Primary Schema list, the Expression field, and wildcards. You can also include or exclude tables that begin with the text you specify.
4. Click **Configure Tables**.
5. In the Configure Table Replication wizard, enter a replicate schema name.
6. Select a replication and materialization action for the selected tables, and click **Execute**.

   At the bottom of the wizard, you see a preview of the selected primary and replicate tables for replication, with their corresponding row count and replicate status.

   The commands are sent to the server as a single batch of requests. The server then executes those commands asynchronously. It may take some time, depending on the number of table requests and the system load for the status to reflect your request.

See also

- *Deploying a Replication Model* on page 12

Monitor Replication and Materialization Table Details

The detailed information for each primary and replicate table includes the requested action, the replication status, and materialization progress. To refresh the current materialization status of the tables, click **Refresh** in the upper right corner.

Column Name	Description
Primary Table Name	Source tables present in the primary database.
Replicate Table Name	Target tables present in the replicate database.
Requested Action	The selected replication and materialization action in the Configure Table Replication wizard.
Status	Status of the table replication.
Materialization Progress	Percentage of materialization progress.
Row Count	The number of rows present in the source table.
Rows Remaining	The number of rows remaining to be replicated to the target table.
Messages	Messages related to the replication and materialization of the target table. Double-click this field to view messages in a dialog.

Configuring DDL Replication

Configure DDL replication in the replication environment.

Prerequisites

Deploy a replication model.

Task

1. In the SAP Replication Management Agent Designer, expand **AgentContainer > RMA**.
2. Right-Click **(REPPATH) reppath**, and select **Setup DDL Replication**.
3. Tick the **Enable DDL replication** check box.
4. Click **Configure** to set the DDL replication configuration.

Monitoring the Replication Environment

Check replication status and other related processes in your environment using RMA.

Viewing Replication Status

View the current status of replication after you have deployed and set up your tables for replication.

1. In the SAP Replication Management Agent Designer, expand **AgentContainer > RMA**.
2. Right-click **(REPPATH) reppath**, and select **Replication Status**.
   The replication status appears in the Replication Status pane on the right.

Obtaining Latency Timings

Send a trace flag to the replication process for instances where you check on the replication status and the latency time is unknown.

1. In the Agent Servers tree view, right-click the **(REPPATH) reppath** node and select **Send a Trace (Ticket)**.

   Note: It may take several seconds for the trace to make its round trip, depending on the system load.

   You see a confirmation message that the trace sent is successful.
2. View the Replication Status to see the replication latency time.

   You may need to refresh the Replication Status by clicking **Refresh**.

Viewing Task Status

Monitor the progress of a recently performed task. Many activities are done asynchronously.

For example, when you mark the tables for replication, a batch of commands is sent to the server and processed one-by-one. You can view the status of this task using RMA.

1. In the SAP Replication Management Agent Designer, expand **AgentContainer**.
2. Right-click the **RMA** node and select **Task Status**.
   You see a Task Status window that shows all the information about the task being performed, along with status messages.
3. (Optional) Click **Refresh** to refresh the status for each task.

Viewing Error Log and Performance Data of the Agent Container

To view the error log and performance data of the Agent Container in real time, use RMA.

1. In the SAP Replication Management Agent Designer, expand the system name in the Agent Servers pane.
2. Select **AgentContainer**.
   The Properties, Server Log, and Performance tabs appear to the right of the window.
3. Click the corresponding error log and performance tabs to see their information.

Viewing Replication Related Log Files

View replication errors, information, and trace messages written to the log files.
Go to the log directory of the replication components to see the log files.

- RMA – `/sybase/`*`SAP_SID`*`_REP/RMA-15_5/instance/AgentContainer/logs`
- SAP Replication Server – `/sybase/`*`SAP_SID`*`_REP/RMA_REP_HANARep/RMA_REP_HANARep.log`
- Replication Agent – `/sybase/`*`SAP_SID`*`_REP/RAX-15_5/RMA_REP_`*`name`*`/log/ RMA_REP_`*`name`*`.log`

Glossary

Glossary of terms used in replication systems.

- **active database –** A database that is replicated to a standby database in a warm standby application. See also *warm standby application*.
- **application programming interface (API) –** A predefined interface through which users or programs communicate with each other. Open Client™ and SAP® Open Server™ are examples of APIs that communicate in a client/server architecture. RCL, the Replication Command Language, is the SAP Replication Server API.
- **applied function –** A replicated function, associated with a function replication definition, that SAP Replication Server delivers from a primary database to a subscribing replicate database. See also *replicated function delivery*, *request function*, and *function replication definition*.
- **article –** A replication definition extension for tables or stored procedures that can be an element of a publication. Articles may or may not contain **where** clauses, which specify a subset of rows that the replicate database receives.
- **asynchronous procedure delivery –** A method of replicating, from a source to a destination database, a stored procedure that is associated with a table replication definition.
- **asynchronous command –** A command that a client submits to SAP Replication Server where the client is not prevented from proceeding with other operations before the completion status is received. Many SAP Replication Server commands function as asynchronous commands within the replication system.
- **atomic materialization –** A materialization method that copies subscription data from a primary to a replicate database through the network in a single atomic operation, using a select operation with a holdlock. No changes to primary data are allowed until data transfer is complete. See also *nonatomic materialization*, *bulk materialization* and *no materialization*.
- **autocorrection –** Autocorrection is a setting applied to replication definitions, using the **set autocorrection** command, to prevent failures caused by missing or duplicate rows in a copy of a replicated table. When autocorrection is enabled, SAP Replication Server converts each update or insert operation into a delete followed by an insert. Autocorrection should only be enabled for replication definitions whose subscriptions use nonatomic materialization.
- **base class –** A function-string class that does not inherit function strings from a parent class. See also *function-string class*.
- **bitmap subscription –** A type of subscription that replicates rows based on bitmap comparisons. Create columns using the int datatype, and identify them as the `rs_address` datatype when you create a replication definition.
- **bulk copy-in –** A feature that improves SAP Replication Server performance when replicating large batches of **insert** statements on the same table in SAP ASE 12.0 and later.

SAP Replication Server implements bulk copy-in in Data Server Interface (DSI), the SAP Replication Server module responsible for sending transactions to replicate databases, using the Open Client™ Open Server™ Bulk-Library.

Bulk copy-in also improves the performance of subscription materialization. When **dsi_bulk_copy** is on, SAP Replication Server uses bulk copy-in to materialize the subscriptions if the number of **insert** commands in each transaction exceeds **dsi_bulk_threshold**.

- **bulk materialization –** A materialization method whereby subscription data in a replicate database is initialized outside of the replication system. You can use bulk materialization for subscriptions to table replication definitions or function replication definitions. For example, data may be transferred from a primary database using media such as magnetic tape, diskette, CDROM, or optical storage disk. Bulk materialization involves a series of commands, starting with define subscription. See also *atomic materialization*, *nonatomic materialization*, and *no materialization*.
- **centralized database system –** A database system where data is managed by a single database management system at a centralized location.
- **class –** See *error class* and *function-string class*.
- **class tree –** A set of function-string classes, consisting of two or more levels of derived and parent classes, that derive from the same base class. See also *function-string class*.
- **client –** A program connected to a server in a client/server architecture. It may be a frontend application program executed by a user or a utility program that executes as an extension of the system.
- **Client/Server Interfaces (C/SI) –** The SAP interface standard for programs executing in a client/server architecture.
- **concurrency –** The ability of multiple clients to share data or resources. Concurrency in a database management system depends upon the system protecting clients from conflicts that arise when data in use by one client is modified by another client.
- **connection –** A connection from an SAP Replication Server to a database. See also *Data Server Interface (DSI)* and *logical connection*.
- **connection profile –** Information required to establish a database connection.
- **coordinated dump –** A set of database dumps or transaction dumps that is synchronized across multiple sites by distributing an rs_dumpdb or rs_dumptran function through the replication system.
- **database –** A set of related data tables and other objects that is organized and presented to serve a specific purpose.
- **database generation number –** Stored in both the database and the RSSD of the SAP Replication Server that manages the database, the database generation number is the first part of the origin queue ID (*qid*) of each log record. The origin queue ID ensures that the SAP Replication Server does not process duplicate records. During recovery operations, you may need to increment the database generation number so that SAP Replication Server does not ignore records submitted after the database is reloaded.

- **database replication definition –** A description of a set of database objects—tables, transactions, functions, system stored procedures, and DDL—for which a subscription can be created.

  You can also create table replication definitions and function replication definitions. See also *table replication definition* and *function replication definition*.
- **database server –** A server program, such as SAP ASE, that provides database management services to clients.
- **data definition language (DDL) –** The set of commands in a query language, such as Transact-SQL, that describes data and their relationships in a database. DDL commands in Transact-SQL include those using the **create**, **drop**, and **alter** keywords.
- **data manipulation language (DML) –** The set of commands in a query language, such as Transact-SQL, that operates on data. DML commands in Transact-SQL include **select**, **insert**, **update**, and **delete**.
- **data server –** A server whose client interface conforms to the SAP Client/Server Interfaces and provides the functionality necessary to maintain the physical representation of a replicated table in a database. Data servers are usually database servers, but they can also be any data repository with the interface and functionality SAP Replication Server requires.
- **Data Server Interface (DSI) –** SAP Replication Server threads corresponding to a connection between an SAP Replication Server and a database. DSI threads submit transactions from the DSI outbound queue to a replicate data server. They consist of a scheduler thread and one or more executor threads. The scheduler thread groups the transactions by commit order and dispatches them to the executor threads. The executor threads map functions to function strings and execute the transactions in the replicate database. DSI threads use an Open Client connection to a database. See also *outbound queue* and *connection*.
- **data source –** A specific combination of a database management system (DBMS) product such as a relational or non-relational data server, a database residing in that DBMS, and the communications method used to access that DBMS from other parts of a replication system. See also *database* and *data server*.
- **decision support application –** A database client application characterized by ad hoc queries, reports, and calculations and few data update transactions.
- **declared datatype –** The datatype of the value delivered to the SAP Replication Server from the Replication Agent:

  - If the Replication Agent delivers a base SAP Replication Server datatype, such as `datetime`, to the SAP Replication Server, the declared datatype is the base datatype.
  - Otherwise, the declared datatype must be the UDD for the original datatype at the primary database.
- **default function string –** The function string that is provided by default for the system provided classes `rs_sqlserver_function_class` and

rs_default_function_class and classes that inherit function strings from these classes, either directly or indirectly. See also *function string*.

- **dematerialization –** The optional process, when a subscription is dropped, whereby specific rows that are not used by other subscriptions are removed from the replicate database.
- **derived class –** A function-string class that inherits function strings from a parent class. See also *function-string class* and *parent class*.
- **direct route –** A route used to send messages directly from a source to a destination SAP Replication Server, with no intermediate SAP Replication Servers. See also *indirect route* and *route*.
- **disk partition –** See *partition*.
- **distributed database system –** A database system where data is stored in multiple databases on a network.
- **Distributor –** An SAP Replication Server thread (DIST) that helps to determine the destination of each transaction in the inbound queue.
- **dump marker –** A message written by SAP ASE in a database transaction log when a dump is performed. In a warm standby application, when you are initializing the standby database with data from the active database, you can specify that SAP Replication Server use the dump marker to determine where in the transaction stream to begin applying transactions in the standby database. See also *warm standby application*.
- **Embedded Replication Server System Database (ERSSD) –** The SAP SQL Anywhere database that stores SAP Replication Server system tables. You can choose whether to store SAP Replication Server system tables on the ERSSD or the SAP ASE RSSD. See also *Replication Server System Database (RSSD)*.
- **Enterprise Connect Data Access (ECDA) –** An integrated set of software applications and connectivity tools that allow access to data within a heterogeneous database environment, such as a variety of LAN-based, non-ASE data sources, and mainframe data sources.
- **error action –** An SAP Replication Server response to a data server error. Possible SAP Replication Server error actions are **ignore**, **warn**, **retry_log**, **log**, **retry_stop**, and **stop_replication**. Error actions are assigned to specific data server errors.
- **error class –** A name for a collection of data server error actions that are used with a specified database.
- **exceptions log –** A set of three SAP Replication Server system tables that holds information about transactions that failed on a data server. The transactions in the log must be resolved by a user or by an intelligent application. You can use the **rs_helpexception** stored procedure to query the exceptions log.
- **ExpressConnect for HANA DB –** A set of libraries that can be used to provide direct communication between SAP Replication Server and an SAP HANA database.
- **ExpressConnect for Oracle –** A set of libraries that can be used to provide direct communication between SAP Replication Server and an Oracle database.

- **Failover –** SAP Failover allows you to configure two version 12.0 and later SAP ASEs as companions. If the primary companion fails, that server's devices, databases, and connections can be taken over by the secondary companion.

  For more detailed information about how SAP Failover works in SAP ASE, refer to *Using SAP Failover in a High Availability System*, which is part of the SAP ASE documentation set.
- **fault tolerance –** The ability of a system to continue to operate correctly even though one or more of its component parts is malfunctioning.
- **function –** An SAP Replication Server object that represents a data server operation such as insert, delete, select, or begin transaction. SAP Replication Server distributes such operations to other SAP Replication Servers as functions. Each function consists of a function name and a set of data parameters. In order to execute the function in a destination database, SAP Replication Server uses function strings to convert a function to a command or set of commands for a type of database. See also *user-defined function*, and *replicated function delivery*.
- **function replication definition –** A description of a replicated function used in replicated function delivery. The function replication definition, maintained by SAP Replication Server, includes information about the parameters to be replicated and the location of the primary version of the affected data. See also *replicated function delivery*.
- **function scope –** The range of a function's effect. Functions have replication definition scope or function-string class scope. A function with replication definition scope is defined for a specific replication definition, and cannot be applied to other replication definitions. A function with function-string class scope is defined once for a function-string class and is available only within that class.
- **function string –** A string that SAP Replication Server uses to map a function and its parameters to a data server API. Function strings allow SAP Replication Server to support heterogeneous replication, in which the primary and replicate databases are different types, with different SQL extensions and different command features.
- **function-string class –** A named collection of function strings used with a specified database connection. Function-string classes include those provided with SAP Replication Server and those you have created. Function-string classes can share function string definitions through function-string inheritance. The three system-provided function-string classes are `rs_sqlserver_function_class`, `rs_default_function_class`, and `rs_db2_function_class`. See also *base class*, *class tree*, *derived class*, *function-string inheritance*, and *parent class*.
- **function-string inheritance –** The ability to share function string definitions between classes, whereby a derived class inherits function strings from a parent class. See also *derived class*, *function-string class*, and *parent class*.
- **function-string variable –** An identifier used in a function string to represent a value that is to be substituted at run time. Variables in function strings are enclosed in question marks (?). They represent column values, function parameters, system-defined variables, or user-defined variables.

- **function subscription –** A subscription to a function replication definition used in both applied and request function delivery.
- **gateway –** Connectivity software that allows two or more computer systems with different network architectures to communicate.
- **generation number –** See *database generation number*.
- **heterogeneous data servers –** Multi-vendor data servers used together in a distributed database system.
- **hibernation mode –** An SAP Replication Server state in which all data definition language (DDL) commands, except admin and sysadmin commands, are rejected; all routes and connections are suspended; most service threads, such as Data Server Interface (DSI) and SAP Replication Server Interface (RSI), are suspended; and RSI and Replication Agent users are logged off and not allowed to log on. This is used during route upgrades, and may be turned on for an SAP Replication Server to debug problems.
- **high-performance analytic appliance (HANA) –** An SAP® in-memory online transaction processing and online analytical processings solution.
- **high-performance analytic appliance database (SAP HANA Database) –** The SAP in-memory database.
- **high availability (HA) –** Very low downtime. Computer systems that provide HA usually provide 99.999% availability, or roughly five minutes unscheduled downtime per year.
- **high volume adaptive replication (HVAR) –** Compilation of a group of **insert**, **delete**, and **update** operations to produce a net result and the subsequent bulk application of the net result to the replicate database.
- **hot standby application –** A database application in which the standby database can be placed into service without interrupting client applications and without losing any transactions. See also *warm standby application*.
- **ID Server –** One SAP Replication Server in a replication system is the ID Server. In addition to performing the usual SAP Replication Server tasks, the ID Server assigns unique ID numbers to every SAP Replication Server and database in the replication system, and maintains version information for the replication system.
- **inbound queue –** A stable queue used to spool messages from a Replication Agent to an SAP Replication Server.
- **indirect route –** A route used to send messages from a source to a destination SAP Replication Server, through one or more intermediate SAP Replication Servers. See also *direct route* and *route*.
- **interfaces file –** A file containing entries that define network access information for server programs in an SAP client/server architecture. Server programs may include SAP ASE, gateways, SAP Replication Servers, and Replication Agents. The interfaces file entries enable clients and servers to connect to each other in a network.
- **latency –** The measure of the time it takes to distribute to a replicate database a data modification operation first applied in a primary database. The time includes Replication Agent processing, SAP Replication Server processing, and network overhead.

- **local-area network (LAN) –** A system of computers and devices, such as printers and terminals, connected by cabling for the purpose of sharing data and devices.
- **locator value –** The value stored in the `rs_locater` table of the SAP Replication Server RSSD that identifies the latest log transaction record received and acknowledged by the SAP Replication Server from each previous site during replication.
- **logical connection –** A database connection that SAP Replication Server maps to the connections for the active and standby databases in a warm standby application. See also *connection* and *warm standby application*.
- **login name –** The name that a user or a system component such as SAP Replication Server uses to log in to a data server, SAP Replication Server, or Replication Agent.
- **Log Transfer Language (LTL) –** A subset of the Replication Command Language (RCL). A Replication Agent such as RepAgent uses LTL commands to submit to SAP Replication Server the information it retrieves from primary database transaction logs.
- **Log Transfer Manager (LTM) –** The Replication Agent program for SAP SQL Server. See also *Replication Agent* and *RepAgent thread*.
- **maintenance user –** A data server login name that SAP Replication Server uses to maintain replicate data. In most applications, maintenance user transactions are not replicated.
- **materialization –** The process of copying data specified by a subscription from a primary database to a replicate database, thereby initializing the replicate table. Replicate data can be transferred over a network, or, for subscriptions involving large amounts of data, loaded initially from media. See also *atomic materialization*, *bulk materialization*, *no materialization*, and *nonatomic materialization*.
- **materialization queue –** A stable queue used to spool messages related to a subscription being materialized or dematerialized.
- **missing row –** A row missing from a replicated copy of a table but present in the primary table.
- **mixed-version system –** A replication system containing SAP Replication Servers of different software versions that have different capabilities based on their different software versions and site versions. Mixed-version support is available only if the system version is 11.0.2 or greater.

  For example, a replication system containing SAP Replication Servers version 11.5 or later and version 11.0.2 is a mixed-version system. A replication system containing SAP Replication Servers of releases earlier than release 11.0.2 is not a mixed-version system, because any newer SAP Replication Servers are restricted by the system version from using certain new features. See also *site version* and *system version*.
- **more columns –** Columns in a replication definition exceeding 250, but limited to 1024. More columns are supported by SAP Replication Server version 12.5 and later.
- **multi-site availability (MSA) –** Methodology for replicating database objects—tables, functions, transactions, system stored procedures, and data definition language (DDL) statements from the primary to the replicate database. See also *database replication definition*.

- **Multi-Path Replication™ –** SAP Replication Server feature that improves performance by enabling parallel paths of data from the source database to the target database. You can configure multi-path replication in warm standby and multisite availability (MSA) environments. These multiple paths process data independently of each other and are applicable when sets of data can be processed in parallel without transactions consistency requirements between them while still maintaining data consistency within a path, but not adhering to the commit order across different paths.
- **name space –** The scope within which an object name must be unique.
- **nonatomic materialization –** A materialization method that copies subscription data from a primary to a replicate database through the network in a single operation, without a holdlock. Changes to the primary table are allowed during data transfer, which may cause temporary inconsistencies between replicate and primary databases. Data is applied in increments of ten rows per transaction, which ensures that the replicate database transaction log does not fill. Nonatomic materialization is an optional method for the **create subscription** command. See also *autocorrection*, *atomic materialization*, *no materialization*, and *bulk materialization*.
- **network-based security –** Secure transmission of data across a network. SAP Replication Server supports third-party security mechanisms that provide user authentication, unified login, and secure message transmission between SAP Replication Servers.
- **no materialization –** A materialization method that lets you create a subscription when the subscription data already exists at the replicate site. Use the **create subscription** command with the **without materialization** clause. You can use this method to create subscriptions to table replication definitions. See also *atomic materialization* and *bulk materialization*.
- **online transaction processing (OLTP) application –** A database client application characterized by frequent transactions involving data modification (inserts, deletes, and updates).
- **Origin Queue ID (qid) –** Formed by the Replication Agent, the qid uniquely identifies each log record passed to the SAP Replication Server. It includes the date and timestamp and the database generation number. See also *database generation number*.
- **orphaned row –** A table row that is present in the replicate, but not in the primary database.
- **outbound queue –** A stable queue used to spool messages. The DSI outbound queue spools messages to a replicate database. The RSI outbound queue spools messages to a replicate SAP Replication Server.
- **parallel DSI –** Configuring a database connection so that transactions are applied to a replicate data server using multiple Data Server Interface (DSI) threads operating in parallel, rather than a single DSI thread. See also *connection* and *Data Server Interface (DSI)*.
- **parameter –** An identifier representing a value that is provided when a procedure executes. Parameter names are prefixed with an @ character in function strings. When a procedure is called from a function string, SAP Replication Server passes the parameter values, unaltered, to the data server. See also *searchable parameter*.

- **parent class –** A function-string class from which a derived class inherits function strings. See also *function-string class* and *derived class*.
- **partition –** A raw disk partition or operating system file that SAP Replication Server uses for stable queue storage. Only use operating system files in a test environment.
- **physical connection –** A connection from an SAP Replication Server to a database.
- **primary data –** The definitive version of a set of data in a replication system. The primary data is maintained on a data server that is known to all of the SAP Replication Servers with subscriptions for the data.
- **primary database –** Any database that contains data that is replicated to another database via the replication system.
- **primary fragment –** A horizontal segment of a table that holds the primary version of a set of rows.
- **primary key –** A set of table columns that uniquely identifies each row.
- **primary site –** The location or facility at which primary data servers and primary databases are deployed to support normal business operations. Sometimes called the active site or main site. See *error class* and *function-string class*.
- **principal user –** The user who starts an application. When using network-based security, SAP Replication Server logs in to remote servers as the principal user.
- **profile –** Allows user to configure a connection with a pre-defined set of properties relative to the server SAP Replication Server is connecting to.
- **projection –** A vertical slice of a table, representing a subset of the table's columns.
- **publication –** A group of articles from the same primary database. A publication lets you collect replication definitions for related tables and/or stored procedures and then subscribe to them as a group. You collect replication definitions as articles in a publication at the source SAP Replication Server and subscribe to them with a publication subscription at the destination SAP Replication Server. See also *article* and *publication subscription*.
- **publication subscription –** A subscription to a publication. See also *article* and *publication*.
- **published datatype –** The datatype of the column after the column-level translation (and before a class-level translation, if any) at the replicate data server. The published datatype must be either an SAP Replication Server base datatype or a UDD for the datatype in the target data server. If the published datatype is omitted from the replication definition, it defaults to the declared datatype.
- **query –** In a database management system, a query is a request to retrieve data that meets a given set of criteria. The SQL database language includes the **select** command for queries.
- **quiescent –** A state in which log scanning has stopped and all scanned records have been propagated to their destinations in a replication system. Some Replication Agent and SAP Replication Server commands require that you first quiesce the replication system.
- **quoted identifiers –** Object names that contain special characters such as spaces and nonalphanumeric characters, start with a character other than alphabet, or correspond to a

reserved word and need to be enclosed in quote (single or double) characters to be parsed correctly.

- **real time loading (RTL) –** High volume adaptive replication (HVAR) to an SAP® IQ database. Uses relevant commands and processes to apply HVAR changes to an SAP IQ replicate database. See *high volume adaptive replication*.
- **remote procedure call (RPC) –** A request to execute a procedure that resides in a remote server. The server that executes the procedure could be an SAP ASE, an SAP Replication Server, or a server created using SAP Open Server. The request can originate from any of these servers or from a client application. The RPC request format is a part of the SAP Client/Server Interfaces.
- **RepAgent thread –** The Replication Agent for SAP ASE databases. Replication Agent is an SAP ASE thread; it transfers transaction log information from the primary database to an SAP Replication Server for distribution to other databases.
- **replicate database –** A database that contains data replicated from another database (the primary database) through a replication system. The replicate database is the database that receives replicated data in a replication system. Contrast with primary database.
- **replicated function delivery –** A method of replicating, from a source to a destination database, a stored procedure that is associated with a function replication definition. See also *applied function*, *request function*, and *function replication definition*.
- **replicated stored procedure –** An SAP ASE stored procedure that is marked as replicated using the **sp_setrepproc** system procedure. Replicated stored procedures can be associated with function replication definitions or table replication definitions. See also *replicated function delivery* and *asynchronous procedure delivery*.
- **replicated table –** A table that is maintained by SAP Replication Server, in part or in whole, in databases at multiple locations. There is one primary version of the table, which is marked as replicated using the **sp_setreptable** system procedure; all other versions are replicated copies.
- **Replication Agent –** A program or module that transfers transaction log information representing modifications made to primary data from a database server to an SAP Replication Server for distribution to other databases. RepAgent is the Replication Agent for SAP ASE databases.
- **Replication Command Language (RCL) –** The commands used to manage information in SAP Replication Server.
- **replication definition –** Usually, a description of a table for which subscriptions can be created. The replication definition, maintained by SAP Replication Server, includes information about the columns to be replicated and the location of the primary version of the table.

  You can also create function replication definitions; sometimes the term “table replication definition” is used to distinguish between table and function replication definitions. See also *function replication definition*.

- **Replication Management Agent (RMA) –** A distributed management agent that you can use to easily set up and manage replication from any supported databases to an SAP HANA database.
- **Replication Server Interface (RSI) –** A thread that logs in to a destination SAP Replication Server and transfers commands from the source SAP Replication Server RSI outbound stable queue to the destination SAP Replication Server. There is one RSI thread for each destination SAP Replication Server that is a recipient of commands from a primary or intermediate SAP Replication Server. See also *outbound queue* and *route*.
- **replication system administrator –** The system administrator that manages routine operations in the Replication Server.
- **Replication Server System Database (RSSD) –** The SAP ASE database containing an SAP Replication Server system tables. The user can choose whether to store SAP Replication Server system tables on SAP ASE or embedded in an SAP SQL Anywhere database hosted by SAP Replication Server. See also *Embedded Replication Server System Database (ERSSD)*.
- **Replication Server system Adaptive Server –** The SAP ASE with the database containing an SAP Replication Server system tables.
- **replication system –** A data processing system where data is replicated in multiple databases to provide remote users with the benefits of local data access. Specifically, a replication system that is based upon SAP Replication Server and includes other components such as Replication Agents and data servers.
- **replication system domain –** All replication system components that use the same ID Server.
- **request function –** A replicated function, associated with a function replication definition, that SAP Replication Server delivers from a primary database to a replicate database. The function passes parameter values to a stored procedure that is executed at the replicate database. The stored procedure is executed at the replicate site by the same user as it is at the primary site. See also *replicated function delivery*, *request function*, and *function replication definition*.
- **resync marker –** When you restart Replication Agent in resync mode, Replication Agent sends the resync database marker to SAP Replication Server to indicate that a resynchronization effort is in progress. The resync marker is the first message Replication Agent sends before sending any SQL data definition language (DDL) or data manipulation language (DML) transactions.
- **route –** A one-way message stream from a source Replication Server to a destination Replication Server. Routes carry data modification commands (including those for RSSDs) and replicated functions or stored procedures between Replication Servers. See also *direct route* and *indirect route*.
- **route version –** The lower of the site version numbers of the route’s source and destination SAP Replication Servers. The supported SAP Replication Server versions use the route version number to determine which data to send to the replicate site. See also *site version*.

- **row migration –** The process whereby column value changes in rows in a primary version of a table cause corresponding rows in a replicate version of the table to be inserted or deleted, based on comparison with values in a subscription's **where** clause.
- **SAP Adaptive Server Enterprise (SAP ASE) –** The SAP version 11.5 and later relational database server. If you choose the RSSD option when configuring SAP Replication Server, SAP ASE maintains SAP Replication Server system tables in the RSSD database.
- **SAP Replication Server –** The SAP server program that maintains replicated data, typically on a LAN, and processes data transactions received from other SAP Replication Servers on the same LAN or on a WAN.
- **schema –** The structure of the database. DDL commands and system procedures change system tables stored in the database. Supported DDL commands and system procedures can be replicated to standby databases when you use SAP Replication Server version 11.5 or later and SAP ASE version 11.5 or later.
- **searchable column –** A column in a replicated table that can be specified in the **where** clause of a subscription or article to restrict the rows replicated at a site.
- **searchable parameter –** A parameter in a replicated stored procedure that can be specified in the **where** clause of a subscription to help determine whether or not the stored procedure should be replicated. See also *parameter*.
- **secondary truncation point –** An SAP ASE database that holds primary data has an active truncation point, marking the transaction log location where SAP ASE has completed processing. This is the primary truncation point.
- **site –** An installation consisting of, at minimum, an SAP Replication Server, data server, and database, and possibly a Replication Agent, usually at a discrete geographic location. The components at each site are connected over a WAN to those at other sites in a replication system. See also *primary site*.
- **site version –** The version number for an individual SAP Replication Server. Once the site version has been set to a particular level, the SAP Replication Server enables features specific to that level, and downgrades are not allowed. See also *software version*, *route version*, and *system version*.
- **software version –** The version number of the software release for an individual SAP Replication Server. See also *site version* and *system version*.
- **SQL Server –** The SAP relational database pre-11.5 server.
- **SQL statement replication –** The process in which the SAP Replication Server receives the SQL statement that modified the primary data, rather than the individual row changes from the transaction log. SAP Replication Server applies the SQL statement to the replicated site. RepAgent sends both the SQL Data Manipulation Language (DML) and individual row changes. Depending on your configuration, SAP Replication Server chooses either individual row change log replication or SQL statement replication.
- **Stable Queue Manager (SQM) –** A thread that manages the stable queues. There is one Stable Queue Manager (SQM) thread for each stable queue accessed by the SAP Replication Server, whether inbound or outbound.
- **Stable Queue Transaction (SQT) interface –** A thread that reassembles transaction commands in commit order. A Stable Queue Transaction (SQT) interface thread reads

from inbound stable queues, puts transactions in commit order, then sends them to the Distributor (DIST) thread or a DSI thread, depending on which thread required the SQT ordering of the transaction.

- **stable queues –** Store-and-forward queues where SAP Replication Server stores messages destined for a route or database connection. Messages written into a stable queue remain there until they can be delivered to the destination SAP Replication Server or database. SAP Replication Server builds stable queues using its disk partitions. See also *inbound queue*, *outbound queue*, and *materialization queue*.
- **standalone mode –** An SAP Replication Server mode used for initiating recovery operations.
- **standby database –** In a warm standby application, a database that receives data modifications from the active database and serves as a backup of that database. See also *warm standby application*.
- **stored procedure –** A collection of SQL statements and optional control-of-flow statements stored under a name in an SAP ASE database. Stored procedures supplied with SAP ASE are called system procedures. Some stored procedures for querying the RSSD are included with the SAP Replication Server software.
- **subscription –** A request for SAP Replication Server to maintain a replicated copy of a table, or a set of rows from a table, in a replicate database at a specified location. You can also subscribe to a function replication definition, for replicating stored procedures.
- **subscription dematerialization –** The optional process, when a subscription is dropped, whereby specific rows that are not used by other subscriptions are removed from the replicate database.
- **subscription materialization –** The process of copying data specified by a subscription from a primary database to a replicate database, thereby initializing the replicate table. Replicate data can be transferred over a network, or, for subscriptions involving large amounts of data, loaded initially from media.
- **subscription migration –** The process whereby column value changes in rows in a primary version of a table cause corresponding rows in a replicate version of the table to be inserted or deleted, based on comparison with values in a subscription's where clause.
- **SAP® Control Center for Replication –** A Web-based solution for monitoring the status and availability of servers in a replication environment.
- **symmetric multiprocessing (SMP) –** On a multiprocessor platform, the ability of an application's threads to run in parallel. SAP Replication Server supports SMP, which can improve server performance and efficiency.
- **synchronous command –** A command that a client submits where the client is prevented from proceeding with other operations before the completion status is received.
- **system function –** A function that is predefined and part of the SAP Replication Server product. Different system functions coordinate replication activities, such as **rs_begin**, or perform data manipulation operations, such as **rs_insert**, **rs_delete**, and **rs_update**.
- **system-provided classes –** SAP Replication Server provides the error class `rs_sqlserver_error_class` and the function-string classes `rs_sqlserver_function_class`, `rs_default_function_class`, and

`rs_db2_function_class`. Function strings are generated automatically for the system-provided function-string classes and for any derived classes that inherit from these classes, directly or indirectly. See also *error class* and *function-string class*.

- **system version –** The version number for a replication system that represents the version for which new features are enabled, for SAP Replication Servers of release 11.0.2 or earlier, and below which no SAP Replication Server can be downgraded or installed. For an SAP Replication Server version 11.5, your use of certain new features requires a site version of 1150 and a system version of at least 1102. See also *mixed-version system*, *site version*, and *software version*.
- **table replication definition –** Identifies a primary table and marks in order for SAP Replication Server to replicate its contents when inserted, updated or deleted. It 'publishes' the data in the publish-subscribe methodology used by SAP Replication Server.
- **table subscription –** A subscription to a table replication definition.
- **thread –** A process running within SAP Replication Server. Built upon SAP Open Server, SAP Replication Server has a multi-threaded architecture. Each thread performs a certain function such as managing a user session, receiving messages from a Replication Agent or another SAP Replication Server, or applying messages to a database. See also *Data Server Interface (DSI)*, *Distributor*, and *Replication Server Interface (RSI)*.
- **transaction –** A mechanism for grouping statements so that they are treated as a unit: either all statements in the group are executed or no statements in the group are executed.
- **Transact-SQL –** The relational database language used with SAP ASE. It is based on standard Structured Query Language (SQL), with Sybase extensions.
- **truncation point –** An SAP ASE database that holds primary data has an active truncation point, marking the transaction log location where SAP ASE has completed processing. This is the primary truncation point.
- **user-defined function –** A function that allows you to create custom applications that use SAP Replication Server to distribute replicated functions or asynchronous stored procedures between sites in a replication system. In replicated function delivery, a user-defined function is automatically created by SAP Replication Server when you create a function replication definition.
- **variable –** See *function-string variable*.
- **version –** *mixed-version system*

  See *mixed-version system*, *site version*, *software version*, and *system version*.
- **warm standby application –** An application that employs SAP Replication Server to maintain a standby database for a database known as the active database. If the active database fails, SAP Replication Server and client applications can switch to the standby database.
- **wide-area network (WAN) –** A system of local-area networks (LANs) connected together with data communication lines.
- **wide columns –** Columns in a replication definition containing `char`, `varchar`, `binary`, `varbinary`, `unichar`, `univarchar`, or Java `inrow` data that are wider that 255 bytes.

- **wide data –** Wide data rows, limited to the size of the data page on the data server. SAP ASE supports page sizes of 2K, 4K, 8K, and 16K.
- **wide messages –** Messages larger that 16K that span blocks.