



Quick Start Guide

SAP Replication Server®
Options 15.7.1 SP200

Microsoft Windows, UNIX, and Linux

DOCUMENT ID: DC00766-01-1571200-01

LAST REVISED: March 2014

Copyright © 2014 by SAP AG or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries. Please see <http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark> for additional trademark information and notices.

Contents

CHAPTER 1: Conventions	1
CHAPTER 2: Before You Begin	5
Additional Replication Resources	5
Server Entries to the Interfaces Files	9
Adding Server Entries to the Interfaces Files	9
Creating the Sybase User Account	10
CHAPTER 3: Oracle Replication Setup	11
System Architecture	11
Prerequisites	12
Planning Your Installation	12
System Requirements	13
Identify the Installation Directory	13
Replication Agent Installation Directory	14
Accessing and Downloading Products	14
Installing Replication Components	15
Installing Replication Server	15
Verifying the Installation	19
Installing the Oracle Instant Client Libraries	19
Installing Replication Agent	21
Verifying the Installation	23
Configuring Replication Components	23
Configuring Replication Server	24
Configuring Replication Server for Replication to the Target Data Server	24
Configuring Replication Server for Replication from the Primary Database	25

- Configuring ExpressConnect for Oracle28
- Configuring Replication Agent30
 - Primary Oracle Database Configuration for Replication30
 - Creating the Replication Agent Instance36
 - Verifying the Replication Agent Instance Installation41
 - Initializing the Replication Agent Instance42
- Testing Replication44**
 - Resetting the Primary Oracle Database for Replication45

- CHAPTER 4: Microsoft SQL Server Replication Setup47**
 - System Architecture47**
 - Prerequisites47**
 - Planning Your Installation48**
 - System Requirements48
 - Identify the Installation Directory48
 - Replication Agent Installation Directory49
 - Accessing and Downloading Products49
 - Installing Replication Components50**
 - Installing ECDA Option for ODBC50
 - Installing Replication Server53
 - Verifying the Installation56
 - Installing Replication Agent57
 - Verifying the Installation58
 - Configuring Replication Components58**
 - Configuring ECDA Option for ODBC59
 - Configuring the Microsoft SQL Server ODBC Driver59
 - DirectConnect Server Configuration and Startup59

Creating a Maintenance User for Replication in Microsoft SQL Server	61
Configuring Replication Server	62
Configuring Replication Server for Replication to the Target Data Server	62
Configuring Replication Server for Replication from the Primary Database	63
Configuring Replication Agent	67
Primary Microsoft SQL Server Configuration and Verification	67
Configuring and Validating the Replication Agent Resource File	71
Initialization	74
Subsequent Initialization	76
Verifying the Replication Agent Replicating State	77
Testing Replication	78
Resetting the Primary Microsoft SQL Server Database for Replication	79
CHAPTER 5: DB2 UDB Replication Setup	81
System Architecture	81
Prerequisites	81
Planning Your Installation	82
System Requirements	82
Identify the Installation Directory	83
Replication Agent Installation Directory	83
Accessing and Downloading Products	83
Installing Replication Components	84
Installing ECDA Option for ODBC	84
Installing Replication Server	87
Verifying the Installation	91
Installing Replication Agent	91
Verifying the Installation	93

Configuring Replication Components	94
Setting DB2 UDB Environment Variables	94
Configuring ECDA Option for ODBC	95
Creating a Maintenance User for Replication in DB2 UDB	95
Configuring Replication Server	96
Configuring Replication Server for Replication to the Target Data Server	96
Configuring Replication Server for Replication from the Primary Database	97
Configuring Replication Agent	101
Primary DB2 UDB Configuration for Replication	102
Creating the Replication Agent Instance	104
Verifying the Replication Agent Instance Installation	108
Initializing the Replication Agent Instance	109
Testing Replication	110
Resetting the Primary DB2 UDB Database for Replication	111
 CHAPTER 6: User Management	 113
 CHAPTER 7: Glossary	 117
 Index	 125

CHAPTER 1 Conventions

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

Style conventions

Key	Definition
monospaced (fixed-width)	<ul style="list-style-type: none">• SQL and program code• Commands to be entered exactly as shown• File names• Directory names
<i>italic monospaced</i>	In SQL or program code snippets, placeholders for user-specified values (see example below).
<i>italic</i>	<ul style="list-style-type: none">• 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 sans serif	<ul style="list-style-type: none">• 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 double 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

Replication Agent™ is a generic term used to describe the Replication Agents for SAP® Adaptive Server® Enterprise (SAP® ASE), Oracle, Microsoft SQL Server, and IBM DB2 for Linux, Unix and Windows. The specific names are:

- RepAgent – Replication Agent thread for SAP Adaptive Server Enterprise
- Replication Agent for Oracle

- Replication Agent for Microsoft SQL Server
- Replication Agent for IBM DB2 UDB

CHAPTER 2 Before You Begin

Review these topics before you install and configure the replication system components.

Additional Replication Resources

While setting up the basic replication system, you may need to refer to additional documents, which are a part of the SAP® Replication Server® documentation.

See <http://infocenter.sybase.com>.

Table 1. Additional References

For Information About	Topic	See
Licensing	Download replication components from the SAP Service Marketplace (SMP).	http://service.sap.com
	Obtain a separate license for each SAP Replication Server and SAP Replication Server Options component.	<ul style="list-style-type: none"> • <i>Replication Server 15.7.1 SP200 > Release Bulletin Replication Server 15.7.1 SP200 for UNIX and Linux > Product Summary > Separately Licensed Products</i> • <i>Replication Server 15.7.1 SP200 > Release Bulletin Replication Server 15.7.1 SP200 for Windows > Product Summary > Separately Licensed Products</i>
	Obtain a separate license for each Replication Server Options component.	<i>Replication Server 15.7.1 SP200 Release Bulletin for Windows > Product Summary > Separately Licensed Products > Replication Server Options</i>

For Information About	Topic	See
Installing components from other sources	Download and install the Oracle instant client libraries.	<i>Replication Server 15.7.1 SP200 > Quick Start Guide for SAP HANA Database > Oracle-to-SAP HANA Database Replication Setup > Installing Replication Components > Installing the Oracle Instant Client Libraries</i>
	Install ODBC libraries for ExpressConnect for HANA database.	<i>Replication Server 15.7.1 SP200 > Release Bulletin Replication Server 15.7.1 SP200 for UNIX and Linux > Installation and Upgrade > Special Installation Instructions > Installing ODBC Libraries for ExpressConnect for HANA Database</i>
Connectivity to other servers	Learn about the interfaces file.	<i>Replication Server 15.7.1 SP200 > Administration Guide Volume 1 > Replication Server Technical Overview > Establish Replication Server Connections > Interfaces File</i>
	Learn how to edit an interfaces file.	<i>Replication Server 15.7.1 SP200 > Quick Start Guide for SAP HANA Database > ASE-to-SAP HANA Database Replication Setup > Installing Replication Components > Installing Replication Server > Editing the Interfaces File</i>
Roles, logins, and permissions	Learn about user roles and Replication Server tasks.	<i>Replication Server 15.7.1 SP200 > Administration Guide: Volume 1 15.7.1 SP200 > Introduction to Replication Server > Replication Server Roles and Responsibilities</i>
	Replication Server, Replication Agent, and database users.	<i>Chapter 6, User Management on page 113</i>

For Information About	Topic	See
Subscriptions	Subscriptions example.	<i>Replication Server 15.7.1 SP200 > Design Guide 15.7.1 SP200 > Implementation Strategies > Corporate Rollup > Subscriptions</i>
	Learn about the materialization process and the types of subscription materialization.	<i>Replication Server 15.7.1 SP200 > Heterogeneous Replication Guide 15.7.1 SP200 > Replication System Overview > Heterogeneous Replication Limitations > Subscription Materialization</i>
	Troubleshoot problems that arise if a subscription is not in a valid state or if the check subscription command returns an error message.	<i>Replication Server 15.7.1 SP200 > Quick Start Guide for SAP HANA Database > Best Practices and Troubleshooting > Troubleshooting Subscription Problem</i>
Configuration parameter reference	Replication Agent configuration parameter list.	<i>Replication Agent 15.7.1 SP200 Reference Manual for Linux, UNIX, and Microsoft Windows > Configuration Parameters > Configuration Parameter Reference</i>
Truncation log management	A RepAgent retrieves transactions from the Adaptive Server primary database log and sends them to the Replication Server.	<i>Replication Server 15.7.1 SP200 > Troubleshooting Guide 15.7.1 SP200 > Adaptive Server Log Problems</i>
	Replication Agent provides features for automatic and manual log truncation.	<i>Replication Agent 15.7.1 SP200 Administration Guide for Linux, UNIX, and Microsoft Windows > Administration > Replication Agent Management > Transaction Log Truncation</i>

For Information About	Topic	See
Status monitoring	Learn how to verify Replication Server thread status.	<i>Replication Server 15.7.1 SP200 > Administration Guide: Volume 2 15.7.1 SP200 > Performance Tuning > Multiprocessor Platforms > Commands to Monitor Thread Status</i>
IBM DB2-specific considerations	Review the general issues and considerations that are specific to using Replication Agent for IBM DB2 UDB.	<i>Replication Agent 15.7.1 SP200 Primary Database Guide for Linux, UNIX, and Microsoft Windows > Replication Agent for UDB > IBM DB2-Specific Considerations</i>
Troubleshooting	Review the procedures you can use to isolate replication failures.	<i>Replication Server 15.7.1 SP200 > Troubleshooting Guide 15.7.1 SP200 > Troubleshooting Overview > Troubleshooting Replication Failures</i>
	Troubleshoot replication failures if Replication Server is down.	<i>Replication Server 15.7.1 SP200 Troubleshooting Guide > Troubleshooting Overview > Troubleshooting Replication Failures > Replication Server Is Down</i>
	Troubleshoot errors that can occur when the Replication Agent starts up.	<i>Replication Server 15.7.1 SP200 > Troubleshooting Guide 15.7.1 SP200 > RepAgent Problems > Problems when Starting the RepAgent</i>
	Troubleshoot Replication Agent failure.	<i>Replication Agent 15.7.1 SP200 Administration Guide for Linux, UNIX, and Microsoft Windows > Troubleshooting > Replication Agent Failure</i>
	Review the basic troubleshooting procedures for Replication Agent and the replication system.	<i>Replication Agent 15.7.1 SP200 Administration Guide for Linux, UNIX, and Microsoft Windows > Troubleshooting</i>

Server Entries to the Interfaces Files

An interfaces file contains network definitions for each Replication Server and data server in a replication system.

The interfaces file is located in `$$SYBASE/interfaces` on UNIX platforms, or `$$SYBASE%\ini\sql.ini` on Microsoft Windows platforms.

When you install Replication Server in an existing installation directory, the installer adds Replication Server information to the existing interfaces file. When you install Replication Server in a unique installation directory, the installer creates a new interfaces file. In this situation, there are two interfaces files—one for your existing SAP applications and one for Replication Server.

The interfaces file requires entries for these components:

- Replication Server
- RSSD Adaptive Server or ERSSD SQL Anywhere for this Replication Server
- Replicate database server

If you use direct load materialization, the interfaces file requires an entry for the Replication Agent location.

For each server, you need:

- Server name
- Host name
- Port number

Adding Server Entries to the Interfaces Files

To add entries for the primary and replicate database server in the SAP Replication Server interfaces file, use **dsedit**.

1. Navigate to `$$SYBASE%\OCS-15_0\bin` (Windows), or `$$SYBASE/OCS-15_0/bin` (UNIX).

2. On Windows, double-click **dsedit.exe**.

On UNIX, enter:

```
dsedit
```

3. Select **InterfacesDriver** and click **OK**.

4. Select **Server Object > Add**.

5. Enter the name of the primary database server. Click **OK**.

6. Enter the host name and port number of the primary database server. For example:

```
chaucer, 35356
```

7. Click **OK**.
8. Select **Server Object > Add**.
9. Enter the name of the replicate database server. Click **OK**.
10. Enter the host name and port number of the replicate database server. For example:
`johnson, 9756`
11. Click **OK** twice, then select **File > Exit**.

Creating the Sybase User Account

Create a Sybase® user account to ensure that Sybase product files and directories are created with consistent ownership and privileges.

You must create a Sybase user account before you install the replication system components. A user, such as the Sybase system administrator—who has read, write, and execute privileges—must perform all installation and configuration tasks. For Windows, the user must have the Administrator privilege.

1. To create a Sybase system administrator account, choose an existing account, or create a new account and assign a user ID, group ID, and password for it.
This account is sometimes called the “sybase” user account. See your operating system documentation for instructions on creating a new user account.
If you have already installed other Sybase software, the “sybase” user already exists.
2. Verify that you can log in to the machine using this account.

Implement a sample replication system for Oracle.

Note: Use the tasks in the *SAP Replication Server Options Quick Start Guide* to set up replication only for proof-of-concept or testing, not for production. Only basic Oracle features are addressed in the example scenario. In particular, Oracle Automatic Storage Management (ASM) and Real Application Clusters (RAC) are not addressed in this guide, although Replication Agent does support these features. See the *Replication Agent Primary Database Guide*.

The replication of sequences is not discussed in the *SAP Replication Server Options Quick Start Guide*. To set up sequence replication, see the *Replication Agent Installation Guide* and the *Replication Agent Primary Database Guide*.

See also

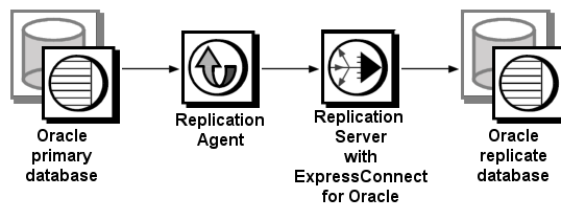
- *Planning Your Installation* on page 12
- *Installing Replication Components* on page 15
- *Configuring Replication Components* on page 23

System Architecture

There are multiple components required to implement a replication system for Oracle.

- A primary Oracle data server
- A replicate Oracle data server
- An SAP Replication Server
- The SAP® Replication Server® Options components:
 - Replication Agent
 - ExpressConnect for Oracle

Figure 1: SAP replication system for Oracle



Prerequisites

Review the prerequisites before you begin setting up replication for Oracle.

Make sure:

- You have an understanding of SAP replication and are familiar with Oracle data servers.
- You have valid licenses for Replication Server and Replication Server Options.
- Oracle database is installed and configured to serve as the source (primary) database from which Replication Agent replicates transactions and as the target (replicate) database to which Replication Server with ExpressConnect for Oracle replicates transactions.
- Install the correct Oracle JDBC thin driver.

Primary Database	Install
Oracle 12c	Oracle JDBC thin driver <code>ojdbc7.jar</code>
Oracle 11g Release 2	Oracle JDBC thin driver 11.2 for JDK 1.6 Note: Install the <code>xstreams.jar</code> for Oracle XStream JDBC driver in the CLASSPATH environment variable.
Oracle 10g or 11g Release 1	Oracle JDBC thin driver for Oracle 10g or 11g and for JDK 1.4 and 1.5

- TCP/IP connectivity is available.
- Any operating system patches required for Java have been installed.
- The Replication Agent has direct access to the primary Oracle online and archived redo logs.

Do not use Replication Server reserved words for object names or connection names. A complete list of reserved words is in the *Replication Server Reference Manual*.

See also

- *Setting Up an Oracle XStream JDBC Driver* on page 31

Planning Your Installation

Before installing the replication components, verify the system requirements, identify the installation directories, and obtain the installation software.

System Requirements

Make sure the target computer on which you are installing the replication components meets the minimum memory and disk space requirements.

For Windows:

Component	Memory (RAM)	Disk space
Replication Server	512MB	380MB
Replication Agent	512MB	350MB

For UNIX:

Component	Memory (RAM)	Disk space
Replication Server	512MB	520MB
Replication Agent	512MB	525MB

For Linux:

Component	Memory (RAM)	Disk space
Replication Server	512MB	400MB
Replication Agent	512MB	525MB

To improve performance by avoiding multiple network hops, install Replication Server, ExpressConnect for Oracle, and the target database on the same machine. Replication Agent must have direct access to the Oracle online and archived redo logs and must be installed on the same machine as the primary database.

See the *Replication Server Options Release Bulletin*.

Identify the Installation Directory

Determine the directory in which to install the replication components.

Replication Agent, Replication Server, and ExpressConnect for Oracle are installed in the same base directory, which is identified by the SYBASE environment variable. Select a path on a host drive to be the recipient of the installation and configuration activities. Replication Agent must be installed on the same type of host as the primary Oracle instance and must be able to directly access the primary Oracle redo and archived redo logs.

Replication Agent Installation Directory

Replication Agent versions 15.5 and later use the RAX-15_5 directory name. If you have an existing Replication Agent installation of version 15.5 or later and do not want to overwrite it, install the latest version of Replication Agent in a different location.

Accessing and Downloading Products

Download replication components from SAP Service Marketplace (SMP) or the Sybase Product Download Center (SPDC).

- For SMP:
 1. Go to the SAP Web site <https://websmp204.sap-ag.de/patches> (login required).
 2. Select **SAP Software Download Center > Installations and Upgrades > A-Z Index** and click **R** for SAP Replication Server products.
 3. Click the Replication Server software you need by version and platform, and follow the instructions.
- For SPDC:

Note: If you are installing Replication Server Options (RSO), this task requires you to perform several steps twice, as you are downloading Replication Server and other products.

1. Go to <https://sybase.subscribenet.com/control/sybs/login>, and enter your login ID and password.
2. Select **Replication Server**.
3. From the high-level Product Information page, select **Replication Server** again.
4. If you are installing SAP® Replication Server®, heterogeneous edition, skip to Step 7. If you are installing RSO, select the Replication Server software you need by version and platform.
The Software Terms and Conditions page appears, listing countries and regions.
5. Either:
 - Select **I Agree**, go to the Product Download Page, and continue with step 6, or
 - Select your country or region to read the terms.
The software licence agreement for the software appears. Read the terms and either:
 - Select **I Agree** to accept the terms, go to the Product Download Page, and continue with step 6, or
 - Select **Cancel** to return to the high-level Product Information page.
6. The Product Download Page lists the software and documentation for the product. Select **FTP Download** or **Advanced Download Options** for the Replication Server software to download and save the compressed files to your local drive.

Note: For each product, the documentation download contains installation-related documents, and the rest of the documentation set can be accessed online.

7. Return to the detailed Product Information page that you accessed in step 3 and select:
 - The RSO software by primary database, version, and platform, or
 - The RSHE software
8. Navigate the Software Terms and Conditions pages as you did in step 5.
9. From the Product Download Page, select **FTP Download** or **Advanced Download Options** for the components to download, and save the compressed files to your local drive.

For RSO, download:

 - Replication Agent 15.7.1
 - ExpressConnect for Oracle 15.7.1

For RSHE, download:

 - Replication Server 15.7.1
 - Replication Agent 15.7.1
 - ExpressConnect for Oracle 15.7.1
10. Uncompress and extract all the installation images to your local drive.

Installing Replication Components

Review the procedures for installing the replication components.

See also

- *Configuring Replication Components* on page 23

Installing Replication Server

Install Replication Server using the setup program.

Prerequisites

- Allocate a disk partition of at least 20MB for each Replication Server you are installing. You can add more partitions later, if necessary. Make sure each partition is available and has write permissions.
- Allocate the entire partition to the Replication Server. If you allocate only a portion of the partition to Replication Server, you cannot use the remainder for any other purpose.
- The sample Replication Server instance `SAMPLE_RS` used in these procedures must be created when you install Replication Server. If you have already installed Replication Server, see the instructions for how to set up a Replication Server instance after installation in *Replication Server ASE-to-ASE Replication Quick Start Guide > SAMPLe_RS*.

Task

1. Verify that the drive on which you install the Replication Server has enough disk space for the components being installed, and at least 100MB of extra disk space for the installation program.
2. Download and extract the Replication Server installation image from the Sybase Product Download Center (SPDC) or the SAP® Service Marketplace (SMP).
3. Start the installation.

- On Windows, launch the **setup** program.

If the installer does not start automatically, double-click `setup.exe` or select **Start > Run** and enter the following:

```
setup.exe
```

If there is not enough disk space in the temporary disk space directory, set the environment variable `TMP` to *directory_name* before running it again, where *directory_name* is the full path to and name of the temporary directory to which the installation program writes the temporary installation files.

- (UNIX or Linux) If you downloaded the product from SPDC, go to the directory where you extracted the installation image and start the installer:

```
./setup.bin
```

where `setup.bin` is the executable file name for installing Replication Server.

If there is not enough disk space in the temporary disk space directory, set the environment variable `IATEMPDIR` to *tmp_dir* before running the installer again, where *tmp_dir* is the full path to the temporary directory to which the installation program writes the temporary installation files.

4. In the Introduction window, click **Next**.
5. Specify where to install Replication Server:

Option	Description
Click Choose .	To select an installation directory. A window appears, and from that window browse for and select the directory.
Enter a new directory path.	To create a new directory.
Click Restore Default Folder .	To restore to the default directory if you do not want to use the directory that you have entered.

- If the directory you chose does not exist, click **Yes** to create it.

- If it does exist, and already contains a Replication Server installation, you are warned that you will overwrite the older version. Click **Next**.

6. Select the type of installation:

Option	Description
Typical	Installs the default components. This is recommended for most users.
Full	Installs every Replication Server component, including all the supported language modules.
Custom	Lets you select the components to install. Some components are automatically installed if they are required to run your selected components.

Click **Next**.

7. Select the geographic location, agree to the license agreement, then click **Next**.
8. On the SySAM License Server window:

Option	Choose
Specify license keys	<p>Either:</p> <ul style="list-style-type: none"> • Click Browse to select the license file. • Use Shift+Click or Ctrl+Click to select multiple license files. The license pane displays the license information. • Copy and paste the license information directly in the license pane. <p>Click Next.</p> <p>If you specify a served license key, the installer prompts you to install a new SySAM license server. Select:</p> <ul style="list-style-type: none"> • Next – to install the new SySAM license server. Follow the installation prompts. • Previous – and select Use previously deployed license server if you have an existing SySAM license server on the same host.
Use previously deployed license server	<p>Enter the:</p> <ul style="list-style-type: none"> • Host name of the machine where the license server is running • Port number if the port number you are using is not the default <p>Click Next.</p>

Option	Choose
Continue installation without a license key	<p>If you do not have licenses for any of the Replication Server components, select this option, then click Next to proceed.</p> <p>The installer allows you to install and use Replication Server components without a license for a grace period of 30 days. To continue using these components after the end of the grace period, obtain valid licenses, and install these licenses using the Replication Server License Installer.</p>

Use **sysadmin lmconfig** to configure and show license management-related information in Replication Server. See *Replication Server Reference Manual > Replication Server Commands > sysadmin lmconfig*.

9. Set SySAM e-mail configuration if necessary. Click **Next**.
10. The installation summary window displays the selections you have made. Review the information, and click **Install**.
11. In the Start Sample Replication Server window, select:

Option	Description
Yes	<p>Note: You cannot start the sample Replication Server on Linux on POWER because the ERSSD requires SAP® SQL Anywhere® Server, which is not available on Linux on POWER.</p> <p>To configure and start a sample Replication Server. The installer displays the configuration information for the sample Replication Server. Record this information.</p> <p>You must enter and confirm a password from 6 to 15 characters in length.</p>
No	<p>To manually configure a full-featured Replication Server and start a sample Replication Server after installation. The installer prompts you to either create a sample Replication Server directory or to proceed with the installation. Select either:</p> <ul style="list-style-type: none"> • Yes – to create a sample Replication Server directory without starting the sample Replication Server, or, • No – to continue with the installation.

Click **Next**.

12. When the installation is complete, click **Done**.

Verifying the Installation

If you have configured and started a sample Replication Server, verify that Replication Server has been successfully installed.

Prerequisites**Task**

1. In a command window, change to the directory where you installed Replication Server.
2. Set the environment variables by sourcing the `SYBASE.csh` file (UNIX or Linux) or by executing the `SYBASE.bat` file (Windows).
3. Log in to Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
```

4. Verify the Replication Server version:

```
admin version
go
```

5. Exit the `isql` session with the `quit` command.

Installing the Oracle Instant Client Libraries

As of version 15.7.1, ECO does not include Oracle Instant Client libraries in its packaging nor installation. Download and install these Instant Client libraries for the installed ECO from the Oracle Web site. ECO and the Oracle Instant Client libraries are loaded by Replication Server and are required for replication to an Oracle target database.

1. Create a temporary directory `<tempdir>`.
2. Go to the Oracle Web site at <http://www.oracle.com>.
3. Select **Downloads > Databases > Instant Client**.
4. Select the download link for your platform:

Table 2. Instant Client Download Links By Platform

Platform	Download Link
Windows x86 32-bit	Instant Client for Microsoft Windows (32-bit)
Windows x86-64 64-bit	Instant Client for Microsoft Windows (64-bit)
Linux x86-64 64-bit	Instant Client for Linux x86-64
Linux pSeries 64-bit	Instant Client for Linux on Power (64-bit)
Solaris SPARC 64-bit	Instant Client for Solaris Operating System (SPARC) (64-bit)

Platform	Download Link
Solaris x86 64-bit	Instant Client for Solaris x86-64
HP-UX Itanium 64-bit	Instant Client for HP-UX Itanium (64-bit)
IBM AIX pSeries 64-bit	Instant Client for AIX5L (64-bit)

5. Read and accept the **OTN Development and Distribution License Agreement for Instant Client**.
6. Go to the specified version and download the Instant Client libraries package to the temporary directory <tempdir>:

Table 3. Instant Client Libraries Packages By Platform

Platform	Instant Client Version	Package
Windows x86 32-bit	10.2.0.4	instantclient-basic-win32-10.2.0.4.zip
Windows x86-64 (64-bit) running Windows-XP, Windows Server 2000, or Windows Server 2003	10.2.0.5	instantclient-basic-win64-10.2.0.5.zip
Windows x86-64 (64-bit) running Windows Server 2008, Windows Vista, or Windows 7	10.2.0.5	instantclient-basic-win64-10.2.0.5.zip
Linux x86-64 64-bit	10.2.0.4	basic-10.2.0.4.0-linux-x86_64.zip
Linux pSeries 64-bit	10.2.0.4	basic-10.2.0.4.0-linux-ppc64.zip
Solaris SPARC 64-bit	10.2.0.4	basic-10.2.0.4.0-solaris-sparc64.zip
Solaris x86 64-bit	10.2.0.4	basic-10.2.0.4.0-solaris-x86-64.zip
HP-UX Itanium 64-bit	10.2.0.4	basic-10.2.0.4.0-hpux-ia64.zip
IBM AIX pSeries 64-bit	10.2.0.4	basic-10.2.0.4.0-aix-ppc64.zip

Note: You must use the exact version of the Instant Client libraries listed in the table. Other versions of the Instant Client libraries are not compatible with ExpressConnect for Oracle.

7. Extract the Oracle Instant Client package into the temporary directory <tempdir> using a zip utility (for Windows) or the **UnZip** utility (for UNIX platforms.)
For UNIX platforms, the UnZip utility can be downloaded from: <https://updates.oracle.com/unzips/unzips.html>.
8. Copy the extracted Oracle Instant Client library files from the <tempdir> \instantclient_10_2 directory to the <eco_install_dir> \REP-15_5\connector\lib directory.

Installing Replication Agent

Install Replication Agent using the GUI wizard.

Note: If there is not enough disk space in your default temporary directory, set the IATEMPDIR (Linux or UNIX) or TMP (Windows) environment variable to a directory that has enough space. Include the full path to this directory.

1. Log in to the Replication Agent host machine using an operating system user account with authority to start, stop, and administer the Replication Agent instance (for example, the “sybase” user).
2. Close all nonessential applications, and minimize any open windows.
3. Download and extract the Replication Agent installation image from the SAP Service Marketplace (SMP).
4. Start the installation program:
 - On Microsoft Windows platforms, the installation program should start automatically. If it does not, start the installation program from Microsoft Windows Explorer by double-clicking the **setup.exe** file.
 - On UNIX platforms, enter:


```
./setup.bin
```
5. Click **Next**.

Note: You can click **Cancel** to stop the installation at any point.

6. Select the geographic location, agree to the license agreement, then click **Next**.
7. Specify an installation directory. The default installation directory is:
 - Existing `%SYBASE%` or `c:\sybase` on Microsoft Windows platforms
 - Existing `$$SYBASE` or `/opt/sybase` on UNIX platforms

Click **Next** to accept the default installation directory, or:

- Click **Choose** to select an installation directory in the file browser. Click **Next**.
- Enter a directory name. Click **Next**.

If you continue and the older products were not installed with the current version of Replication Agent, the installation program overwrites any common files.

If the directory name does not exist, click **Yes** when prompted to create it:

CHAPTER 3: Oracle Replication Setup

```
The directory does not exist.  
Do you want to create it?
```

If the directory you specified exists, you see:

```
Warning: You have chosen to install into an existing  
directory. If you proceed with this installation,  
any older versions of the products you choose to  
install that are detected in this directory will be  
replaced.
```

Note: On Microsoft Windows platforms, if you are prompted to overwrite a DLL, click **Yes** only if the version of the new DLL is newer than the one the installation program is attempting to overwrite.

8. Select the type of database the Replication Agent will connect to.

Click **Next**.

9. Select one of these options to enter the license:

- **Specify license keys** – browse to or specify the license file.
- **Use previously deployed license server** – use a previously deployed license server. Enter the host name of the machine where the license server is running and the port number if the port number you are using is not the default.
- **Continue installation without a license key** – install and use Replication Agent without a license for a grace period of 30 days. To continue using Replication Agent after the end of the grace period, obtain a valid license from the SAP Service Marketplace (SMP), and install it.

Note: Replication Agent is licensed statically when installed with RSHE. If you are installing Replication Agent with RSHE, locate the file that contains a valid SySAM license for the Replication Agent component in RSHE. Specify that license file here.

See the *SySAM Users Guide*.

Click **Next** until you see the SySAM Notification window.

10. On the SySAM Notification window, configure your server for e-mail notification. When configuration is enabled, you receive information about license management events that require attention. Select **Yes** and either accept the default values that are supplied, or enter values for:
 - SMTP server host name
 - SMTP server port number
 - E-mail return address
 - Recipients of the notification
 - Message severity level of an event that triggers e-mail notification:
 - Informational
 - Warning
 - Error

If you choose not to have e-mail alerts or severity messages logged, select **No**.

Click **Next**.

11. Review the product features or components listed on the Preinstallation Summary window. Click **Install**.

The installation program installs the components in the installation directory you specified, and displays an installation progress indicator.

If errors occur during the installation, the installation program displays error messages. Exit the installation program wizard to correct the cause of the error, then restart the installation program.

If the software is installed successfully, you see a window confirming the successful installation.

12. Click **Done** to complete the installation and close the installation program.

Verifying the Installation

Verify that Replication Agent has been successfully installed.

1. In a command window, change to the directory where you installed Replication Agent.
2. Set the environment variables by sourcing the `SYBASE.csh` file (UNIX or Linux) or by executing the `SYBASE.bat` file (Windows).
3. Change to `$(SYBASE)/RAX-15_5/bin` (UNIX or Linux) or `%SYBASE%\RAX-15_5\bin` (Windows).
4. Obtain the Replication Agent version string:

- For Windows:

```
ra.bat -v
```

- For UNIX or Linux:

```
./ra.sh -v
```

Note: Make sure permission for the `ra.sh` file is set no broader than 700 (read/write/execute for user, no permissions for group and other).

If Replication Agent installs successfully, you see the SAP copyright and the Replication Agent version string.

Configuring Replication Components

Review the procedures for configuring each of the replication components in the replication system.

See also

- *Testing Replication* on page 44

Configuring Replication Server

Review the procedures for configuring Replication Server for Oracle.

Configuring Replication Server for Replication to the Target Data Server

Create a Replication Server connection to the replicate data server.

1. Navigate to the `$SYBASE/RAX-15_5/scripts/oracle` (UNIX or Linux) or `%SYBASE%\RAX-15_5\scripts\oracle` (Windows) directory.
2. Make a copy of the `oracle_create_rs_standby_connection.sql` script named `my_oracle_create_rs_standby_connection.sql`.
3. Before executing the `my_oracle_create_rs_standby_connection.sql` script against your Replication Server, change all occurrences of `{rds}`, `{rdb}` to the name of the connection that Replication Server uses to connect to ExpressConnect for Oracle where:
 - `rds` is the replicate Oracle System ID (SID) name.
 - `rdb` is any valid identifier. SAP recommends that you use the replicate Oracle SID name.
 - `maintuser` and `password` are the user name and password created in ExpressConnect for Oracle.

For example:

```
create connection to OracleSID_name.OracleSID_name
using profile rs_oracle_to_oracle;eco
set username maintuser
set password "password"
go
```

Note: In the previous example, **password** is a Replication Server reserved word and therefore, must be enclosed in double quotes.

If you are using an ExpressConnect for Oracle connection profile, **dsi_proc_as_rpc** is enabled by default.

4. For configuring the server connectivity for ExpressConnect for Oracle, copy the `tnsnames.ora` file to the `\RS_installation_directory\REP-15_5\connector\oraoci\network\admin` directory.
5. Create the connection to the replicate database:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i
my_oracle_create_rs_standby_connection.sql
```

6. Log in to `SAMPLE_RS`, and verify the Replication Server connection to the standby database:

```
isql -Usa -Psa-pass -SSAMPLE_RS
admin who
go
```

You should see a message returned for the DSI connection *{rds}.{rdb}*. Verify that the status is “Awaiting Message” or “Awaiting Command.”

- Exit the **isql** session with the **quit** command.

Configuring Replication Server for Replication from the Primary Database

Configure Replication Server for replication from the primary database by granting create object permission to the **rs_username** user.

The **rs_username** user, which is used for communication between Replication Agent and Replication Server, must have **create object** permission before you can use it to create a replication definition. You must grant this permission manually from Replication Server.

- Log in to Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
```

If you have not set up the sample Replication Server instance, enter your Replication Server instance name in place of SAMPLE_RS.

- Enter:

```
grant create object to SAMPLE_RS_ra
go
```

where *SAMPLE_RS_ra* is the value for the **rs_username** parameter you entered in the resource file.

Creating a Replication Server Connection to the Primary Database

Create a Replication Server connection to the primary database.

- Go to the `$SYBASE/RAX-15_5/scripts/oracle` (UNIX or Linux) or `%SYBASE%\RAX-15_5\scripts\oracle` directory (Windows).
- Make a copy of the `oracle_create_rs_primary_connection.sql` script.

```
cp oracle_create_rs_primary_connection.sql
my_oracle_create_rs_primary_connection.sql
```

- Before executing the `my_oracle_create_rs_primary_connection.sql` script against your Replication Server, change all occurrences of *{pds}*.*{pdb}* to the name of the Replication Server connection used for the connection from Replication Agent for Oracle, where:

- pds* is the name of the primary Oracle server.
- pdb* is the primary Oracle System ID (SID).

For example, *NY.NYora92*.

Note: Save these values to be used later for the **rs_source_ds** and **rs_source_db** parameters in the Replication Agent `oracle.rs` file.

4. Change *sys* and *sys_pwd* to the user ID and password of the Oracle user who must have permission to apply DML operations against all user tables to be replicated, where:
 - *sys* is the user ID of the Oracle user (**sys**).
 - *sys_pwd* is the password of the Oracle user (**change_on_install**).

Note: This user must be a valid user in the Oracle database, and it must be different from the user defined by the Replication Agent **pds_username** parameter, which is used for Replication Agent connections to the primary database.

This example creates a Replication Server connection to the primary database:

```
create connection to NY.NYora92
set error class rs_oracle_error_class
set function string class rs_oracle_function_class
set username sys
set password change_on_install
with log transfer on, dsi_suspended
go
```

5. Execute the script in Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i
<script_path>\my_oracle_create_rs_primary_connection.sql
```

where *<script_path>* is the full path to
my_oracle_create_rs_primary_connection.sql.

You see a message indicating that the Replication Server connection to the primary database has been created.

See also

- *Granting create object Permission to the rs_username User* on page 26
- *Creating a Database Replication Definition* on page 27
- *Creating a Database Replication Subscription* on page 28

Granting create object Permission to the rs_username User

Grant create object permission to the user ID specified by the Replication Agent **rs_username** parameter.

The **rs_username** user, which is used for communication between Replication Agent and Replication Server, must have **create object** permission before it can be used to create a replication definition. You must grant this permission manually from Replication Server.

1. Log in to Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
```

2. Enter:

```
grant create object to SAMPLE_RS_ra
go
```


where `SAMPLE_RS_ra` is the value for the `rs_username` parameter you entered in the resource file.

3. Connect to the RSSD:

```
1>connect
2>go
```

4. Verify that the `rs_username` user has `create object` permission:

```
1>rs_helpuser
2>go
```

If `rs_username` has `create object` permission, the permission is displayed. For example:

```
SAMPLE_RS_ra connect source, create object
```

See also

- *Creating a Database Replication Definition* on page 27
- *Creating a Database Replication Subscription* on page 28

Creating a Database Replication Definition

Create a database replication definition.

1. Navigate to the `%SYBASE%\RAX-15_5\scripts\sybase` (Windows) or `$SYBASE/RAX-15_5/scripts/sybase` (UNIX or Linux) directory.
2. Make a copy of the `rs_create_test_db_repdef.sql` script named `my_rs_create_db_repdef.sql`.

```
cp rs_create_test_db_repdef.sql
my_rs_create_db_repdef.sql
```

3. Before executing the `my_rs_create_db_repdef.sql` script, change all occurrences of `{pds}`.`{pdb}` to the name of the connection string you defined for the primary database, where:
 - `pds` is the name of the primary Oracle server.
 - `pdb` is the primary Oracle System ID (SID).

For example, `NY.NYora92`.

Note: Save these values to be used later for the `rs_source_ds` and `rs_source_db` parameters in the Replication Agent `oracle.rs` file.

This is a database replication definition example:

```
create database replication definition NY_repdef1
with primary at NY.NYora92
replicate DDL
go
```

4. Connect to the Replication Server and enter:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i my_rs_create_db_repdef.sql
```

You see a message indicating that the database replication definition has been created.

See also

- *Creating a Database Replication Subscription* on page 28

Creating a Database Replication Subscription

Create a database replication subscription.

1. Make a copy of the `rs_create_test_db_sub.sql` script named `my_rs_create_db_sub.sql`.
2. Before executing the `my_rs_create_db_sub.sql` script, change all occurrences of `{pds}`, `{pdb}` and `{rds}`, `{rdb}` to the appropriate connection name, where:
 - `pds` is the name of the primary Oracle server.
 - `pdb` is the primary Oracle System ID (SID).
 - `rds` is the replicate Oracle SID.
 - `rdb` is any valid identifier. SAP recommends that you use the replicate Oracle SID.

Note: Save the values of `pds` and `pdb` to be used later for the `rs_source_ds` and `rs_source_db` parameters in the Replication Agent `oracle.rs` file.

Here is an example of how to create a database replication subscription:

```
create subscription NY_sub1
for database replication definition NY_repdef1
with primary at NY.NYora92
with replicate at orcl1.oratest2
without materialization
go
```

3. Connect to the Replication Server and enter:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i my_rs_create_db_sub.sql
```

You see a message indicating that the subscription is being created.

Configuring ExpressConnect for Oracle

Configure ExpressConnect for Oracle to set up connections between Replication Server and Oracle.

Prerequisites

Make sure that you have installed the Oracle Instant Client libraries before proceeding.

Task

1. Copy the Oracle server `tnsnames.ora` file to the `RS_installation_directory\REP-15_5\connector\oraoci\network\admin` directory.
2. Determine the Oracle user ID and password used to connect from Replication Server. See “Oracle replicate database permissions” in the *Replication Server 15.7.1 Heterogeneous Replication Guide*.
3. Restart Replication Server.
4. Using **isql** create a connection to Replication Server using the alias name defined in the Oracle `tnsnames.ora` file, user ID, and password combination. For example:

```
create connection to
<tnsnames_alias>.<ora_rdb_name>
using profile rs_oracle_to_oracle;eco
set username <userid>
set password <password>
set batch to 'off'
```

where:

- **tnsnames_alias** is the case-sensitive name identifying the replicate Oracle database in the `tnsnames.ora` file. For example:

```
<tnsnames_alias> =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = hostname) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = orcl)
    )
  )
```

- **ora_rdb_name** can be any name which best describes the replicate Oracle database. For example, `orcl11g`.

If you are not creating the connection using one of the Replication Server connection profiles for ExpressConnect for Oracle, set **dsi_proc_as_rpc** to `on` in the **create connection** command. For example:

```
create connection to <tnsnames_alias>.<ora_rdb_name>
set error class <error_class>
set function string class <function_class>
set username <userid>
set password <password>

set batch to 'off'
set dsi_proc_as_rpc to 'on'
```

If you are using one of the ExpressConnect for Oracle connection profiles, **dsi_proc_as_rpc** is enabled by default.

See *Replication Server Reference Manual* for more information on the **create connection** command. For more information about configuring ExpressConnect for Oracle, see the *ExpressConnect for Oracle Configuration Guide*.

Configuring Replication Agent

Review the procedures for configuring Replication Agent for Oracle.

Primary Oracle Database Configuration for Replication

Configure the primary Oracle database for replication and to ensure that the Oracle redo log data is not overwritten before it is read by Replication Agent.

In addition to Oracle LogMiner, you can now alternatively configure the Replication Agent for Oracle Log Reader component to use Oracle XStream APIs to read from the transaction logs. The XStream APIs enable you to share data changes between Oracle and non-Oracle data sources, such as SAP Replication Server.

Note: You can configure the Replication Agent for Oracle Log Reader component to use XStream APIs from version 15.7.1 SP120 and later.

Verifying the Current Archive Setting of the Redo Logs

Verify that redo logging, which Oracle uses to maintain a log infrastructure, is turned on in the primary database.

1. Use SQLPLUS to connect to Oracle as a system administrator.
2. From SQLPLUS, run:

```
select log_mode from v$database;
```

- If the archive log is on, you should see:

```
LOG_MODE  
-----  
ARCHIVELOG
```

3. To turn on log archiving:

```
shutdown;  
startup mount;  
alter database archivelog;  
alter database open;
```

See the *Replication Agent Primary Database Guide*.

Verifying the Supplemental Logging of Primary Key Data

By default, Oracle does not log primary keys and unique indexes. For successful replication of all table values, you must log these values.

1. Use SQLPLUS to connect to Oracle as a system administrator:

```
SELECT SUPPLEMENTAL_LOG_DATA_MIN,  
SUPPLEMENTAL_LOG_DATA_PK,
```

```
SUPPLEMENTAL_LOG_DATA_UI
FROM V$DATABASE;
```

If logging of primary key and unique index values is enabled, the return values are:

```
SUP SUP SUP
--- --- ---
YES YES YES
```

2. If the result is different, turn on supplemental logging by executing:

```
ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (PRIMARY
KEY, UNIQUE INDEX) COLUMNS;
ALTER DATABASE ADD SUPPLEMENTAL LOG DATA;
```

Configuring Oracle LogMiner

Configure Oracle LogMiner on the primary Oracle database.

1. If you have not already installed Oracle LogMiner, go to \$ORACLE_HOME/rdbms/admin, log in as a “sys as sysdba” user, and execute the Oracle LogMiner installation script:

```
@dbmslm.sql
```

2. If the public synonym DBMS_LOGMNR does not exist, create a public synonym so that you do not have to log in as the owner to execute LogMiner functions:

```
CREATE PUBLIC SYNONYM DBMS_LOGMNR FOR
SYS.DBMS_LOGMNR;
```

Setting Up an Oracle XStream JDBC Driver

To use the Replication Agent for Oracle with XStream APIs, use the **setenv** function to set or modify ORACLE_HOME, LD_LIBRARY_PATH, and CLASSPATH environment variables.

Prerequisites

- Download and install the appropriate XStream JDBC driver from the Oracle Web site for your primary Oracle database environment in which Replication Agent resides.
- Make sure your Java Runtime Environment (JRE) is compatible with your Oracle database for using XStream APIs. For example, if you have installed 64-bit Oracle, use the 64-bit JRE 1.7.x. See the Oracle documentation.

By default, Replication Agent uses 64bit JRE when creating an instance with the XStream APIs Log Reader component.

Task

1. Set the ORACLE_HOME environment variable to the Oracle installation directory for your platform:

```
setenv $ORACLE_HOME <install_dir>/oracle/<version>/
```

CHAPTER 3: Oracle Replication Setup

where *install_dir* is the installation directory and *version* is the version number of the Oracle database.

The Oracle installation directory might be different, based on the database version installed on your platform. Set the `ORACLE_HOME` variable before setting the `LD_LIBRARY_PATH` and `CLASSPATH` environment variables.

2. Add the path of the Oracle XStream JDBC driver to the `CLASSPATH` environment variable for your platform.

Add only one version of a JDBC driver in the `CLASSPATH`; otherwise, Replication Agent will have problems connecting to the primary database.

On Linux:

```
setenv LD_LIBRARY_PATH $ORACLE_HOME/lib:$LD_LIBRARY_PATH
```

```
setenv CLASSPATH $ORACLE_HOME/jdbc/lib/ojdbc6.jar:$ORACLE_HOME/rdbs/jlib/xstreams.jar:$CLASSPATH
```

On HP-UX64:

```
setenv SHLIB_PATH $ORACLE_HOME/lib:$SHLIB_PATH
```

```
setenv CLASSPATH $ORACLE_HOME/jdbc/lib/ojdbc6.jar:$ORACLE_HOME/rdbs/jlib/xstreams.jar:$CLASSPATH
```

On AIX 64:

```
setenv LIBPATH $ORACLE_HOME/lib:$LIBPATH
```

```
setenv CLASSPATH $ORACLE_HOME/jdbc/lib/ojdbc6.jar:$ORACLE_HOME/rdbs/jlib/xstreams.jar:$CLASSPATH
```

On SUNOS-i86pc:

```
setenv LD_LIBRARY_PATH $ORACLE_HOME/lib:$LD_LIBRARY_PATH
```

```
setenv CLASSPATH $ORACLE_HOME/jdbc/lib/ojdbc6.jar:$ORACLE_HOME/rdbs/jlib/xstreams.jar:$CLASSPATH
```

On SUNOS-64:

```
setenv LD_LIBRARY_PATH_64 $ORACLE_HOME/lib:$LD_LIBRARY_PATH_64
```

```
setenv CLASSPATH $ORACLE_HOME/jdbc/lib/ojdbc6.jar:$ORACLE_HOME/rdbs/jlib/xstreams.jar:$CLASSPATH
```

where:

- *Oracle_Home* – is the Oracle database installation directory.
- *path_name* – is the library path where you installed the JDBC driver:
 - *LD_LIBRARY_PATH* on UNIX
 - *SHLIB_PATH* on HP-UX64
 - *LIBPATH* on AIX
- *driver* – is the name of the JDBC driver; for Oracle 11.2.0.3, `ojdbc6.jar`. The `xstreams.jar` is the name of the XStream library path.

3. (On Windows) Set Oracle environment variables in any of these ways:

Use the Control Panel	<ul style="list-style-type: none"> • Select Start > Settings > Control Panel > System > Advanced System Settings > Environment Variables. • Click New in the User variables pane. • In the New User Variable dialog, enter: <ul style="list-style-type: none"> • Variable name – is the environment variable that you want to set. For example, ORACLE_HOME, PATH, and CLASSPATH. • Variable value – is the directory path that you want to set for the environment variable. For example, c:\oracle\version (installation directory), C:\ORACLE_HOME\bin (the library path), and C:\ORACLE_HOME\jdbc\lib\ojdbc6.jar (the JDBC driver path).
Edit the existing variables	<ul style="list-style-type: none"> • Select the PATH and CLASSPATH environment variables in the System variables pane, and click Edit. • In the Edit System Variable dialog, using the semicolon (;) path separator, enter the variable value for: <ul style="list-style-type: none"> • Path – %ORACLE_HOME%\bin;%ORACLE_HOME%\LIB;%path% • CLASSPATH – %ORACLE_HOME%\jdbc\lib\ojdbc6.jar;%ORACLE_HOME%\rdbms\jlib\xstreams.jar;%CLASSPATH%
In the command line prompt	Enter: <pre>set CLASSPATH=ORACLE_HOME\jdbc\lib \ojdbc6.jar;ORACLE_HOME\rdbms\jlib\xstreams.jar</pre>

Creating Oracle User and Grant Permissions

Create an Oracle database user for use by Replication Agent when connected to the primary database.

Note: Permission to grant access to objects owned by “SYS” may require the command to be executed by an Oracle user with **sysdba** privileges.

1. Use SQLPLUS to connect to the primary database as a system administrator and run these commands to create an Oracle user named “RA_USER,” specified by **pds_username**, with the password “sybase,” and grant permissions to the user:

```
CREATE USER RA_USER PROFILE "DEFAULT" IDENTIFIED
BY "sybase" DEFAULT TABLESPACE "USERS" ACCOUNT
UNLOCK;
GRANT ALTER ANY PROCEDURE TO RA_USER;
GRANT ALTER ANY TABLE TO RA_USER;

GRANT ALTER SESSION TO RA_USER;
GRANT ALTER SYSTEM TO RA_USER;
```

CHAPTER 3: Oracle Replication Setup

```
GRANT CONNECT TO RA_USER;
GRANT CREATE PROCEDURE TO RA_USER;
GRANT CREATE PUBLIC SYNONYM TO RA_USER;
GRANT CREATE SESSION TO RA_USER;
GRANT CREATE TABLE TO RA_USER;
GRANT DROP ANY PROCEDURE TO RA_USER;
GRANT DROP ANY TABLE TO RA_USER;
GRANT DROP PUBLIC SYNONYM TO RA_USER;
GRANT EXECUTE_CATALOG_ROLE TO RA_USER;
GRANT EXECUTE ON SYS.DBMS_FLASHBACK TO RA_USER;
GRANT EXECUTE ON SYS.DBMS_LOCK TO RA_USER;
GRANT RESOURCE TO RA_USER;
GRANT SELECT ANY TRANSACTION TO RA_USER;
GRANT SELECT_CATALOG_ROLE TO RA_USER;
GRANT SELECT ON SYS.ARGUMENT$ TO RA_USER;
GRANT SELECT ON SYS.ATTRIBUTE$ TO RA_USER;
GRANT SELECT ON SYS.CCOL$ TO RA_USER;
GRANT SELECT ON SYS.CDEF$ TO RA_USER;
GRANT SELECT ON SYS.COL$ TO RA_USER;
GRANT SELECT ON SYS.COLLECTION$ TO RA_USER;
GRANT SELECT ON SYS.COLTYPE$ TO RA_USER;
GRANT SELECT ON SYS.CON$ TO RA_USER;
GRANT SELECT ON SYS.DEFERRED_STG$ TO RA_USER;
GRANT SELECT ON SYS.ICOL$ TO RA_USER;
GRANT SELECT ON SYS.IND$ TO RA_USER;
GRANT SELECT ON SYS.INDCOMPART$ TO RA_USER;
GRANT SELECT ON SYS.INDPART$ TO RA_USER;
GRANT SELECT ON SYS.INDSUBPART$ TO RA_USER;
GRANT SELECT ON SYS.LOB$ TO RA_USER;
GRANT SELECT ON SYS.LOBCOMPPART$ TO RA_USER;
GRANT SELECT ON SYS.LOBFrag$ TO RA_USER;
GRANT SELECT ON SYS.MLOG$ TO RA_USER;
GRANT SELECT ON SYS.NTAB$ TO RA_USER;
GRANT SELECT ON SYS.OBJ$ TO RA_USER;
GRANT SELECT ON SYS.OPQTYPE$ TO RA_USER;
GRANT SELECT ON SYS.PARTOBJ$ TO RA_USER;
GRANT SELECT ON SYS.PROCEDUREINFO$ TO RA_USER;
GRANT SELECT ON SYS.RECYCLEBIN$ TO RA_USER;
GRANT SELECT ON SYS.SEG$ TO RA_USER;
GRANT SELECT ON SYS.SEQ$ TO RA_USER;
GRANT SELECT ON SYS.SNAP$ TO RA_USER;
GRANT SELECT ON SYS.TAB$ TO RA_USER;
GRANT SELECT ON SYS.TABCOMPART$ TO RA_USER;
GRANT SELECT ON SYS.TABPART$ TO RA_USER;
GRANT SELECT ON SYS.TABSUBPART$ TO RA_USER;
GRANT SELECT ON SYS.TS$ TO RA_USER;
GRANT SELECT ON SYS.TYPE$ TO RA_USER;
GRANT SELECT ON SYS.USER$ TO RA_USER;
GRANT SELECT ON V_$LOGMNR_CONTENTS TO RA_USER;
GRANT SELECT ON V_$LOGMNR_LOGS TO RA_USER;
```

For Oracle 12c:

```
GRANT LOGMINING TO RA_USER
```


The Replication Agent for Oracle Log Reader that uses XStream APIs to read from Oracle 11g and 12c transaction logs requires

DBMS_XSTREAM_AUTH.GRANT_ADMIN_PRIVILEGE.

To grant the XStream administrator privilege to the **pds_username**, issue:

```
BEGIN
DBMS_XSTREAM_AUTH.GRANT_ADMIN_PRIVILEGE( grantee =>
'pds_username' );
END;
/
```

See the *Replication Agent Primary Database Guide*.

2. If you configure the **ra_admin_owner** user, make sure that these permissions are also granted:

```
GRANT CREATE SESSION TO RA_USER;
GRANT CREATE TABLE TO RA_USER;
GRANT CREATE SEQUENCE TO RA_USER;
GRANT CREATE ANY PROCEDURE TO RA_USER;
GRANT SELECT_CATALOG_ROLE TO RA_USER;
```

Verifying the Oracle User Roles Created by the Replication Agent

Validate the required Oracle settings.

1. Use SQLPLUS to run this command as the new Oracle user:

```
select GRANTED_ROLE from USER_ROLE_PRIVS;
GRANTED_ROLE
-----
CONNECT
RESOURCE
SELECT_CATALOG_ROLE
```

The user who starts the Replication Agent instance must also have read access to the Oracle redo log files and the Oracle archive directory that contains the archive log files to be accessed for replication. If the Replication Agent is configured to remove old archive files, the user must have update authority to the directory and the archive log files.

2. Add the JDBC driver to the CLASSPATH environment variable. For example, you can append the existing CLASSPATH to the new JDBC driver and add this command to the .login script of a UNIX and Linux user:

```
setenv CLASSPATH /path_name/driver.jar:$CLASSPATH
```

where *driver.jar* is your JDBC driver (ojdbc14.jar, ojdbc5.jar, ojdbc6.jar, or ojdbc7.jar).

For Windows, add this to the CLASSPATH environment variable:

```
set CLASSPATH=path_name\jdbc\lib\driver.jar;%CLASSPATH%
```

where *driver.jar* is your JDBC driver (ojdbc14.jar, ojdbc5.jar, ojdbc6.jar, or ojdbc7.jar).

Creating the Replication Agent Instance

A single installation of the Replication Agent can support replication from multiple databases, however, one Replication Agent instance is needed for each Oracle database that is to be replicated.

Prepare and create a Replication Agent instance for each Oracle database being replicated.

1. Obtain the Oracle JDBC driver from Oracle, and update the CLASSPATH environment variable.

To find the JDBC connection that communicates with the primary Oracle instance, locate the JDBC driver that was available when the Oracle release was originally created.

Primary Database	Install
Oracle 12c	Oracle JDBC thin driver <code>ojdbc7.jar</code>
Oracle 11g Release 2	Oracle JDBC thin driver 11.2 for JDK 1.6 Note: Install the <code>xstreams.jar</code> for Oracle XStream JDBC driver in the CLASSPATH environment variable.
Oracle 10g or 11g Release 1	Oracle JDBC thin driver for Oracle 10g or 11g and for JDK 1.4 and 1.5

Replication Agent may require a newer version of the JDBC driver, which can be found at <http://www.oracle.com/technetwork/indexes/downloads/index.html>.

2. Add the JDBC driver to the CLASSPATH environment variable. For example, you can append the existing CLASSPATH to the new JDBC driver and add this command to the .login script of a UNIX and Linux user:

```
setenv CLASSPATH /path_name/driver.jar:$CLASSPATH
```

where *driver.jar* is your JDBC driver (`ojdbc14.jar`, `ojdbc5.jar`, `ojdbc6.jar`, or `ojdbc7.jar`).

For Windows, add this to the CLASSPATH environment variable:

```
set CLASSPATH=path_name\jdbc\lib\driver.jar;%CLASSPATH%
```

where *driver.jar* is your JDBC driver (`ojdbc14.jar`, `ojdbc5.jar`, `ojdbc6.jar`, or `ojdbc7.jar`).

3. Record the location of the `tnsnames.ora` file, and record the connection name from that file for the Replication Agent to use to connect to the Oracle primary database. You can find these values from any machine from which you can use an Oracle SQLPLUS session to connect to the desired Oracle instance. The ORACLE_SID is from the environment variable, \$ORACLE_SID.

Record the ORACLE_SID value for the instance you want to connect to.

4. (Optional) If your operating system has a Java 6.0.x-compatible JRE or JDK installed, you can use the Oracle **iSQL** (the browser-based interface to SQL*PLUS) demo items to connect to Oracle using the JDBC driver and to verify the connection information.

See "Verifying a JDBC Client Installation" in the *Oracle JDBC Developer's Guide and Reference Guide*.

5. Obtain a local copy of the Oracle `timezone` file, so Replication Agent can correctly process the Oracle `timestamp with timezone` datatype.

Note: This step is required only if the Replication Agent is on a machine (host) other than the Oracle host and does not have access to the Oracle `timezone` file.

There are two `timezone` files in the `$ORACLE_HOME/oracore/zoneinfo` (UNIX or Linux) or `%ORACLE_HOME%\oracore\zoneinfo` (Windows) directory:

- The `timezone.dat` binary file is the default file that contains the most commonly used time zones.
- The `timezlg.dat` binary file contains a larger set of defined time zones.

The Oracle instance uses the `timezone` file from the `ORA_TZFILE` environment variable. If `ORA_TZFILE` is not set, use the default `timezone` file.

Make a copy of the `timezone` file available on the machine hosting the Replication Agent. To determine which `timezone` file is used by the primary Oracle instance, connect to the Oracle instance and execute:

```
select * from V$TIMEZONE_FILE
```

Note: These files are Oracle version- and platform-dependent. You cannot use a `timezone` file from a little-endian platform on a big-endian platform, nor can you use a `timezone` file from a version of Oracle that is different than the version of the primary Oracle.

6. Locate the Replication Agent for Oracle resource file template.

The majority of configuration values required to create and initialize a Replication Agent can be recorded and stored in a resource file. Using a resource file provides a means to record or retain the configuration information for a Replication Agent instance, allowing an instance to be removed and re-created.

The resource file template (`oracle.rs`) is in the Replication Agent directory `$SYBASE/RAX-15_5/init` (UNIX or Linux) or `%SYBASE%\RAX-15_5\init` (Windows).

7. Create an instance resource file:

Copy the resource file template `$SYBASE/RAX-15_5/init/oracle.rs` (UNIX or Linux) or `%SYBASE%\RAX-15_5\init\oracle.rs` (Windows) to another file that contains the configuration information for a new instance, for example, `myra.rs`.

CHAPTER 3: Oracle Replication Setup

Change the values assigned to properties in the resource file so they match the values required for your environment. The majority of these values are the location, user, and password information listed in the `tnsname.ora` file for the Oracle, Replication Server, and RSSD connections.

- Set the value of **rs_source_ds** to the value you chose for *pds* when you created a Replication Server connection from the primary database.
- Set the value of **rs_source_db** to the value you chose for *pdb* when you created a Replication Server connection from the primary database.
- When you create a Replication Agent instance with a resource file, the **pds_username** and **pds_password** configuration parameter values must not contain single or double quotes if the **create_pds_username** parameter is set to yes.

The resource file is self-documenting, with each parameter described. See the *Replication Agent Administration Guide*. For a complete list of configuration parameters, see the *Replication Agent Reference Manual*.

Warning! The **rs_source_ds** and the **rs_source_db** values must match the *{pds,pdb}* values of your Replication Server primary connection name that you configured when you created a Replication Server connection from the primary database.

Table 4. Resource File Parameters

Parameter	Description	Example Values
instance name	Any valid name.	myra
admin_port	Port number that Replication Agent uses.	9030 (if in use, select a different port number)
ltm_admin_user	User name for administering the Replication Agent instance.	sa
ltm_admin_pw	Password for the user administering the Replication Agent instance.	The default password policy is that the password must be between 6 and 255 characters.
pds_host_name	Machine (host) where Oracle is installed.	jdoh_host1
pds_port_number	Port number for Oracle.	1521
pds_database_name	\$ORACLE_SID	NYora92
pds_username	User ID that Replication Agent uses to access primary data server.	RA_USER
pds_password	Password for pds_username .	sybase
rs_host_name	Machine where Replication Server is installed.	jdoh_host1
rs_port_number	Port where Replication Server is installed.	11752

Parameter	Description	Example Values
rs_username	Replication Server user with connect source and create object permissions. This is the user used to create a Replication Server connection from the primary database.	SAMPLE_RS_ra
rs_password	Password for rs_username .	sa_pass
rs_charset	Character set that Replication Server is using. Note: The value defined for the rs_charset configuration parameter must match the RS_charset value in the Replication Server configuration file, \$SYBASE/REP-15_5/install/<server>.cfg (UNIX or Linux) or %SYBASE%\REP-15_5\install\<server>.cfg (Windows).	<ul style="list-style-type: none"> Windows: cp850 UNIX: iso_1
rs_source_ds	Valid name representing data server of Oracle primary database. This value is used to create a Replication Server connection from the primary database.	pds
rs_source_db	Valid name representing Oracle primary database. This value is used to create a Replication Server connection from the primary database.	pdb
rssd_host_name	Machine where RSSD resides.	jdoe_host1
rssd_port_number	Port number where RSSD resides.	11751
rssd_database_name	Database name for RSSD.	SAMPLE_RS_ERSSD
rssd_username	Valid user for RSSD.	SAMPLE_RS_RSSD_maint
rssd_password	Password for rssd_username .	sa_pass
pdb_timezone_file	Path of the Oracle timezone file.	/software/oracle/Ora10g/oracore/zoneinfo/timezone.dat
start_instance	Start the instance that was created.	yes
initialize_instance	Initialize the Replication Agent instance.	yes
pdb_include_archives	Enables or disables the use of Oracle archive log files.	USE_DEFAULT

Parameter	Description	Example Values
pdb_archive_path	Identifies directory path where Replication Agent expects to find archived Oracle redo log files.	A valid directory path on the machine hosting Replication Agent that points to a location where Oracle puts the archived redo log files.

8. Create and execute the new instance resource file.

Note: Be sure your CLASSPATH environment variable points to the correct Oracle JDBC driver before proceeding.

a) Validate the settings in the resource file:

- On Windows:

```
%SYBASE%\RAX-15_5\bin\ra_admin.bat -vr myra.rs
```

- On UNIX or Linux:

```
$$SYBASE/RAX-15_5/bin/ra_admin.sh -vr myra.rs
```

where *myra.rs* is the path and name of the resource file.

Note: Make sure permission for the *ra_admin.sh* file is set no broader than 700 (read/write/execute for user, no permissions for group and other).

Validation results are returned as either:

- Response-file processing completed
- Or
- Response-file processing completed with errors

If any validation fails, the **ra_admin** utility returns an error message and information about the failure. You can repeat the validation process as many times as necessary until it executes without error. No entities are changed or created.

b) Create and configure the Replication Agent instance:

- On Windows:

```
%SYBASE%\RAX-15_5\bin\ra_admin.bat -r myra.rs
```

- On UNIX or Linux:

```
$$SYBASE/RAX-15_5/bin/ra_admin.sh -r myra.rs
```

Note: Make sure permission for the *ra_admin.sh* file is set no broader than 700 (read/write/execute for user, no permissions for group and other).

where *myra.rs* is the path and name of the resource file.

Note: If, in your response file, you set **start_instance** to yes, your instance is also running. If you set **initialize_instance** to yes, your instance is also initialized.

c) Results are returned as either:

- Response-file processing completed
- Response-file processing completed with errors

See the *Replication Agent Administration Guide*.

9. If you set **start_instance** to yes in your resource file, and if the log indicates that the Replication Agent instance was started, skip this step, and continue with the next section.

Change to the instance directory and run Replication Agent in the background.

- On Windows:

```
cd %SYBASE%\RAX-15_5\myra
```

- On UNIX or Linux:

```
cd $SYBASE/RAX-15_5/myra
```

Execute the RUN file in the background.

Your Replication Agent for Oracle is now running.

See also

- *Verifying the Replication Agent Instance Installation* on page 41
- *Initializing the Replication Agent Instance* on page 42

Verifying the Replication Agent Instance Installation

Verify that the Replication Agent instance has been successfully installed.

1. Use **dsedit** to update the Replication Server `sql.ini` (Windows) or `interfaces` (UNIX or Linux) file to include an entry for the Replication Agent location.

Note: You can use any Tabular Data Stream™ (TDS) client utility (**isql**, **isqlApp**, or **SQLAdvantage**) that you prefer.

2. Verify the connection to Replication Agent:

- a) Open a command window in the `%SYBASE%` (Windows) or `$SYBASE` (UNIX or Linux) directory of your Replication Agent installation.
- b) Set the environment variables by executing the `SYBASE.bat` file (Windows) or by sourcing the `SYBASE.csh` file (UNIX or Linux).
- c) Log in to Replication Agent:

```
isql -Usa -Psa_pass -Smyra
```

Note: These verification steps are optional because they were performed when you verified the resource file.

3. Verify the Replication Agent connection to Replication Server:

- a) Enter:

```
test_connection RS
go
```

- b) This result is returned:

```
Type Connection
-----
RS succeeded
(1 row affected)
```

- c) If the result indicates a failure, either the server is not responding or the connection properties (*host*, *port*, *user*, or *password*) are incorrect. Verify the host and port configuration values, and manually log in to the Replication Server as the configured user to determine which property is incorrectly defined.

See the *Replication Agent Administration Guide*.

4. Verify the Replication Agent connection to the primary database:

- a) Enter:

```
test_connection PDS
go
```

You see:

```
Type Connection
-----
PDS succeeded
(1 row affected)
```

- b) If the result indicates a failure:

- The server is not responding, or
- One of the connection properties is incorrect.

Check the `tnsnames.ora` file and `tnsnames` configuration values, and manually log in to the primary Oracle database as the configured user to find which property is incorrectly defined.

See *Testing Network Connectivity* in the *Replication Agent Administration Guide*.

See also

- *Initializing the Replication Agent Instance* on page 42

Initializing the Replication Agent Instance

Initialize the Replication Agent instance.

Note: If you set `initialize_instance` to yes in your resource file, and if you received no errors while the instance was created, skip to step 2.

1. The `ra_admin init` command verifies that the primary Oracle database is correctly configured to provide archived logging and supplemental logging, and that the Oracle user ID used by the Replication Agent has the necessary permissions. In addition to creating objects in the database to support stored procedure replication, the `ra_admin init` command also initializes the Replication Agent System Database (RASD) by reading schema information, and redo log location information from the primary Oracle database.

To initialize the Replication Agent instance, run:

```
ra_admin init
go
```

2. Enter:

```
resume
go
```

If the Replication Agent successfully transfers to a replicating state, you see:

```
State                Action
-----
REPLICATING         Ready to replicate data.
```

The Replication Agent goes to the REPLICATION DOWN state if an error occurs:

```
isql -Usa -Psa_pass -Smyra
1> resume
2> go
Msg 32000, Level 20, State 0:
Server 'myra', Procedure 'resume', Line 1:
Command <resume> failed - Desired state <REPLICATE>
could not be achieved. Current state: <REPLICATION DOWN>
```

- 3. The `ra_status` command returns the state of the Replication Agent. It is a good practice to verify that the Replication Agent remains in replication state, even after the resume command executes successfully.**

To detect an error that occurred after replication start-up, execute:

```
ra_status
go
```

If the Replication Agent is in replicating state, `ra_status` returns:

```
State                Action
-----
REPLICATING         Ready to replicate data.
```

The Replication Agent goes to the REPLICATION DOWN state if an error occurs:

```
1> ra_status
2> go
State                Action
-----
REPLICATION DOWN    Waiting for operator command.
```

- 4. Validate that both primary and replicate connections are active:**

```
isql -Usa -Psa_pass -SSAMPLE_RS
admin who
go
```

Note: Be aware that the DSI connection for the primary database connection is usually suspended because you are not replicating data back to the primary database.

Do not proceed until **admin who** returns status for threads similar to this:

CHAPTER 3: Oracle Replication Setup

Spid	Name	State	Info
13	DSI EXEC	Awaiting Command	101(1) SAMPLE_RS_ERSSD.
9	DSI	Awaiting Message	101 SAMPLE_RS_ERSSD.
8	SQM	Awaiting Message	101:0 SAMPLE_RS_ERSSD.
54	DSI EXEC	Awaiting Command	102(1) rds.rdb
53	DSI	Awaiting Message	102 rds.rdb
17	SQM	Awaiting Message	102:0 rds.rdb
20	DSI EXEC	Suspended	103(1) pds.pdb
21	DSI	Suspended	103 pds.pdb
24	DIST	Awaiting Wakeup	103 pds.pdb
25	SQT	Awaiting Wakeup	103:1 DIST pds.pdb
23	SQM	Awaiting Message	103:1 pds.pdb
22	SQM	Awaiting Message	103:0 pds.pdb
62	REP AGENT	Awaiting Command	pds.pdb

Testing Replication

When you finish setting up the replication system, test that replication works as intended.

1. Connect to the primary Oracle instance as a regular user, rather than as the maintenance user. Make sure the regular user also exists in the replicate database.
2. Create a test table to replicate in the primary database:

```
create table T1 (a int, b char(10));
```

3. Grant permissions to any new or existing object to be replicated in the primary database:

```
grant all on T1 to public;
```

4. Connect to the replicate Oracle instance as a regular user, rather than as the maintenance user.

5. Create a test table in the replicate database:

```
create table T1 (a int, b char(10));
```

6. Grant permissions to any new or existing object to be replicated in the replicate database so that the Replication Server maintenance user can update this table:

```
grant all on T1 to public;
```

7. Connect to Replication Agent, and mark the table:

```
pdb_setreptable T1, mark
```

8. After the table is marked and the Replication Agent is in a replicating state, insert test data into the test table and commit it.

By default, the work performed by the maintenance user of the primary connection is not replicated. The user ID used to insert transactions cannot be the same as the maintenance user defined in the primary connection.

Replication Agent applies only committed transactions to a replicate database. However, because Oracle expects a commit command to be issued, follow simple test commands with an explicit **commit** command:

```
insert into T1 values ( 42, 'foo' );
commit;
```

9. Use your preferred query tool to examine the replicate site for results and compare the contents of your test table from both the primary and replicate sites.

Resetting the Primary Oracle Database for Replication

In a test environment, you may occasionally need to reset the replication environment.

1. Log in to the Replication Agent and suspend replication:

```
isql -Usa -Psa_pass -Smyra
suspend
go
```

2. Set the transaction log truncation point to the end of the current transaction log.

```
ra_locator move_truncpt
```

3. Refresh the Replication Agent repository rather than overwriting it:

```
ra_admin refresh
```

Note: If you prefer to delete and replace all the information in the Replication Agent repository, issue the **ra_admin deinit, force** command followed by a normal **ra_admin init** command (without the **force** option).

4. Reset the locator stored in Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
connect
go
rs_zeroltm pds, pdb
go
exit
go
```

You see:

```
Locator has been reset to zero.
(return status = 0)
```

5. Resume replication with the Replication Agent **resume** command:

```
isql -Usa -Psa_pass -Smyra
resume
go
```


Microsoft SQL Server Replication Setup

Implement a sample replication system for Microsoft SQL Server.

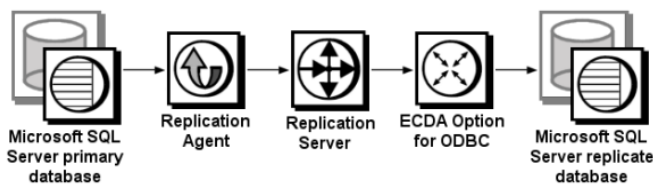
Note: Use the tasks in the *Replication Server Options Quick Start Guide* to set up replication only for proof-of-concept or testing, not for production.

System Architecture

There are multiple components required to implement a replication system for Microsoft SQL Server.

- A primary Microsoft SQL Server database
- A replicate Microsoft SQL Server database
- An SAP Replication Server
- The SAP Replication Server Options components:
 - Replication Agent
 - Enterprise Connect™ Data Access (ECDA) Option for ODBC

Figure 2: SAP replication system for Microsoft SQL Server database



Prerequisites

Review the prerequisites before you begin setting up replication for Microsoft SQL Server.

Make sure:

- You have an understanding of SAP replication and are familiar with Microsoft SQL Server.
- ECDA for ODBC is installed on the same host as your replicate Microsoft SQL Server database.

CHAPTER 4: Microsoft SQL Server Replication Setup

- You have valid licenses for Replication Server and Replication Server Options.
- Microsoft SQL Server 2008, Microsoft SQL Server 2008 R2, or Microsoft SQL Server 2012 is installed and configured to serve as the source (primary) database from which Replication Agent replicates transactions and as the target (replicate) database to which Replication Server with ECDA replicates transactions.
- Microsoft SQL Server JDBC driver 4.0 is installed. This driver is supported for Microsoft SQL Server 2008, Microsoft SQL Server 2008 R2, and Microsoft SQL Server 2012.
- TCP/IP connectivity is available.

Because Replication Agent for Microsoft SQL Server is limited to the Microsoft Windows platform, all examples shown are for Windows.

Note: On Windows Server 2008 and Windows Server 2008 R2, you must be logged in as an Administrator.

Do not use Replication Server reserved words for object names or connection names. A complete list of reserved words is in the *Replication Server Reference Manual*.

Planning Your Installation

Before installing the replication components, verify the system requirements, identify the installation directories, and obtain the installation software.

System Requirements

Make sure the target computer on which you are installing the replication components meets the minimum memory and disk space requirements.

Components	Memory (RAM)	Disk space
Replication Server	512MB	380MB
ECDA	512MB	440MB
Replication Agent	512MB	300MB

To improve performance by avoiding multiple network hops, install Replication Server, ECDA for ODBC, and the target database on the same machine.

See the *Replication Server Options Release Bulletin*.

Identify the Installation Directory

Determine the directory in which to install the replication components.

Replication Agent, Replication Server, and ECDA are installed in the same base directory, which is identified by the SYBASE environment variable. Select a path on the host drive to be the recipient of the installation and configuration activities. Replication Agent must be

installed on a host from which it can directly access the primary Microsoft SQL Server instance transaction log.

Replication Agent Installation Directory

Replication Agent versions 15.5 and later use the RAX-15_5 directory name. If you have an existing Replication Agent installation of version 15.5 or later and do not want to overwrite it, install the latest version of Replication Agent in a different location.

Accessing and Downloading Products

Download replication components from SAP Service Marketplace (SMP) or the Sybase Product Download Center (SPDC).

- For SMP:
 1. Go to the SAP Web site <https://websmp204.sap-ag.de/patches> (login required).
 2. Select **SAP Software Download Center > Installations and Upgrades > A-Z Index** and click **R** for SAP Replication Server products.
 3. Click the Replication Server software you need by version and platform, and follow the instructions.
- For SPDC:

Note: If you are installing Replication Server Options (RSO), this task requires you to perform several steps twice, as you are downloading Replication Server and other products.

1. Go to <https://sybase.subscribenet.com/control/sybs/login>, and enter your login ID and password.
2. Select **Replication Server**.
3. From the high-level Product Information page, select **Replication Server** again.
4. If you are installing Replication Server Heterogeneous Edition (RSHE), skip to Step 7. If you are installing RSO, select the Replication Server software you need by version and platform.
The Software Terms and Conditions page appears, listing countries and regions.
5. Either:
 - Select **I Agree**, go to the Product Download Page, and continue with step 6, or
 - Select your country or region to read the terms.
The software licence agreement for the software appears. Read the terms and either:
 - Select **I Agree** to accept the terms, go to the Product Download Page, and continue with step 6, or
 - Select **Cancel** to return to the high-level Product Information page.
6. The Product Download Page lists the software and documentation for the product. Select **FTP Download** or **Advanced Download Options** for the Replication Server software to download and save the compressed files to your local drive.

Note: For each product, the documentation download contains installation-related documents, and the rest of the documentation set can be accessed online.

7. Return to the detailed Product Information page that you accessed in step 3 and select:
 - The RSO software by primary database, version, and platform, or
 - The RSHE software
8. Navigate the Software Terms and Conditions pages as you did in step 5.
9. Uncompress and extract all the installation images to your local drive.

Installing Replication Components

Review the procedures for installing the replication components.

Install all replication components on the same host where a Microsoft SQL Server data server has already been installed and is running with both the primary and replicate databases. This simplifies the quick-start process and avoids the step of separately installing a Microsoft SQL Server ODBC driver for ECDA Option for ODBC.

See also

- *Configuring Replication Components* on page 58

Installing ECDA Option for ODBC

Install ECDA Option for ODBC using the setup program.

Prerequisites

Install ECDA on the same server as the replicate database to eliminate a network hop and to improve performance.

Task

Note: To install using the console mode or a response file, see the *ECDA Installation Guide for Linux and UNIX* or the *ECDA Installation Guide for Microsoft Windows*.

1. Verify that the drive on which you will install the products has enough free disk space for the products being installed, and at least 10MB extra disk space for the installation program. The program frees this extra space after the installation is complete.
2. Insert the ECDA CD-ROM into the CD-ROM drive.
 - On Microsoft Windows, the installation program should start automatically. If it does not, select **Start > Run**, and enter:

```
x:\setup.exe
```

where *x*: is your CD drive.

- On Linux or UNIX, if auto-mount is not available on your machine, follow local procedures for mounting the CD-ROM, and enter:

```
./setup.bin
```

The **Welcome** window appears.

3. Click **Next**. You may see the following error message:

```
Error writing file = There may not be enough temporary disk
space. Try using -is:tempdir to use a temporary directory on
a partition with more disk space.
```

If so, set the temporary directory to another directory that has more disk space by entering the following at the command line:

- (Windows)

```
x:\setup.exe -is:tempdir <directory_name>
```

- (UNIX or Linux)

```
./setup.bin -is:tempdir <directory_name>
```

where *<directory_name>* is the name of the temporary directory to which the installation program will write its temporary files.

4. Read the License and Copyright Agreement. Using the drop-down list at the top of the window, select the geographic location where you are installing ECDA to display the agreement appropriate to your region. Select **I agree**, and click **Next**.

You must agree to the license and copyright agreement before you can continue.

5. In the installation directory window, click **Next** to accept the default directory for the installation, or enter a different directory where you want to install ECDA. One of the following occurs:

- If the installation directory you chose does not exist, the installation program prompts:
The directory does not exist. Do you want to create it?

Click **Yes**.

- If the installation directory exists and contains files, the software prompts:

```
Warning: You have chosen to install into an existing
directory. If you proceed with this installation,
any older versions of the products you choose to
install that are detected in this directory will be
replaced.
```

Note: You will not see this message if the directory exists but is empty.

If you are prompted to overwrite any files or DDL, select **Yes** only if the version of the new files or DDL is later than the one you are attempting to overwrite.

Note: In certain cases when ECDA is installed with other SAP products, you may see warnings about overwriting newer versions of files. In these cases, simply instruct the

installer to overwrite these files and proceed with the installation by selecting **Yes to All**.

6. Select one of the two installation types:
 - Custom, which allows you to select the ECDA options and components to install.
 - Full, which installs all ECDA products and components from the CD.

Click **Next**.

7. Based on your selection:
 - If you selected Custom, the next window displays all the ECDA options and all the components “checked” or selected. You must “uncheck” or deselect the components that you do not want to install.

Note: Be aware that if you deselect a component that is required, it is automatically installed if it is needed to run other selected components.

After you have selected either the ECDA Option for ODBC or ECDA Option for Oracle and have selected the desired components, click **Next**.

- If you selected Full install, the next window displays all the ECDA options and all the components “checked” or selected. Click **Next**.

Before proceeding to the next window, the installation program verifies the selections, and checks for dependencies and available disk space. The **Product Summary** window displays every component that is to be installed and the total disk space required for all the selected components.

Note: If the target directory does not have enough free space, the space-required and the space-available information is displayed. If you click **Next** without sufficient hard disk space, an error occurs that stops the installation.

8. Verify that you have selected the correct type of installation, and that you have enough disk space to complete the process. Click **Next** to continue the installation.

The installation program unloads all the components from the CD and displays a progress indicator. When the installation is complete, a message appears indicating that the installation program wizard has completed installing your product.

9. Click **Next**. The SySAM License Server window opens and displays this prompt:

Will licenses be obtained from the License Server?

- Select **Yes** if you have a pre-existing SySAM network license server installed or will install a new license server.
Enter the host name and the port number of the license server.
- Select **No** if you do not have a SySAM license server installed and are going to use a local license file. A message appears telling you to download the license file.

If you select **Yes** and the license server is not found, or if you select **No** and the local license file is not installed on your local machine, this message appears:

Installer can't check out a license. Do you want to continue without a license?

10. Click **Yes** to complete the installation. If you select **No**, this message appears:

Please remember to download and install the license file after this installation.

Click **OK**.

The SysAM Notification window asks you to configure your server for e-mail notification. When configuration is enabled, you will receive information about license management events requiring attention.

11. Enter the following:

- SMTP server host name
- SMTP server port number
- E-mail Return Address
- Recipient e-mail addresses
- Message severity that triggers e-mail messages

Click **Next**.

12. A window displays a message indicating that the installation was successful and advising you to check for software updates. Click **Finish**.

Installing Replication Server

Install Replication Server using the setup program.

Prerequisites

- Allocate a disk partition of at least 20MB for each Replication Server you are installing. You can add more partitions later, if necessary. Make sure each partition is available and has write permissions.
- Allocate the entire partition to the Replication Server. If you allocate only a portion of the partition to Replication Server, you cannot use the remainder for any other purpose.
- The sample Replication Server instance SAMPLE_RS used in these procedures must be created when you install Replication Server. If you have already installed Replication Server, see the instructions for how to set up a Replication Server instance after installation in *Replication Server ASE-to-ASE Replication Quick Start Guide > SAMPLE_RS*.

Task

1. Verify that the drive on which you install the Replication Server has enough disk space for the components being installed, and at least 100MB of extra disk space for the installation program.

CHAPTER 4: Microsoft SQL Server Replication Setup

2. Download and extract the Replication Server installation image from the Sybase Product Download Center (SPDC) or the SAP® Service Marketplace (SMP).
3. Start the installation.

- On Windows, launch the **setup** program.

If the installer does not start automatically, double-click `setup.exe` or select **Start > Run** and enter the following:

```
setup.exe
```

If there is not enough disk space in the temporary disk space directory, set the environment variable `TMP` to *directory_name* before running it again, where *directory_name* is the full path to and name of the temporary directory to which the installation program writes the temporary installation files.

- (UNIX or Linux) If you downloaded the product from SPDC, go to the directory where you extracted the installation image and start the installer:

```
./setup.bin
```

where `setup.bin` is the executable file name for installing Replication Server.

If there is not enough disk space in the temporary disk space directory, set the environment variable `IATEMPDIR` to *tmp_dir* before running the installer again, where *tmp_dir* is the full path to the temporary directory to which the installation program writes the temporary installation files.

4. In the Introduction window, click **Next**.
5. Specify where to install Replication Server:

Option	Description
Click Choose .	To select an installation directory. A window appears, and from that window browse for and select the directory.
Enter a new directory path.	To create a new directory.
Click Restore Default Folder .	To restore to the default directory if you do not want to use the directory that you have entered.

- If the directory you chose does not exist, click **Yes** to create it.
 - If it does exist, and already contains a Replication Server installation, you are warned that you will overwrite the older version. Click **Next**.
6. Select the type of installation:

Option	Description
Typical	Installs the default components. This is recommended for most users.
Full	Installs every Replication Server component, including all the supported language modules.
Custom	Lets you select the components to install. Some components are automatically installed if they are required to run your selected components.

Click **Next**.

7. Select the geographic location, agree to the license agreement, then click **Next**.
8. On the SySAM License Server window:

Option	Choose
Specify license keys	<p>Either:</p> <ul style="list-style-type: none"> • Click Browse to select the license file. • Use Shift+Click or Ctrl+Click to select multiple license files. The license pane displays the license information. • Copy and paste the license information directly in the license pane. <p>Click Next.</p> <p>If you specify a served license key, the installer prompts you to install a new SySAM license server. Select:</p> <ul style="list-style-type: none"> • Next – to install the new SySAM license server. Follow the installation prompts. • Previous – and select Use previously deployed license server if you have an existing SySAM license server on the same host.
Use previously deployed license server	<p>Enter the:</p> <ul style="list-style-type: none"> • Host name of the machine where the license server is running • Port number if the port number you are using is not the default <p>Click Next.</p>
Continue installation without a license key	<p>If you do not have licenses for any of the Replication Server components, select this option, then click Next to proceed.</p> <p>The installer allows you to install and use Replication Server components without a license for a grace period of 30 days. To continue using these components after the end of the grace period, obtain valid licenses, and install these licenses using the Replication Server License Installer.</p>

CHAPTER 4: Microsoft SQL Server Replication Setup

Use **sysadmin Imconfig** to configure and show license management-related information in Replication Server. See *Replication Server Reference Manual > Replication Server Commands > sysadmin Imconfig*.

9. Set SySAM e-mail configuration if necessary. Click **Next**.
10. The installation summary window displays the selections you have made. Review the information, and click **Install**.
11. In the Start Sample Replication Server window, select:

Option	Description
Yes	<p>Note: You cannot start the sample Replication Server on Linux on POWER because the ERSSD requires SAP® SQL Anywhere® Server, which is not available on Linux on POWER.</p> <hr/> <p>To configure and start a sample Replication Server. The installer displays the configuration information for the sample Replication Server. Record this information.</p> <p>You must enter and confirm a password from 6 to 15 characters in length.</p>
No	<p>To manually configure a full-featured Replication Server and start a sample Replication Server after installation. The installer prompts you to either create a sample Replication Server directory or to proceed with the installation. Select either:</p> <ul style="list-style-type: none">• Yes – to create a sample Replication Server directory without starting the sample Replication Server, or,• No – to continue with the installation.

Click **Next**.

12. When the installation is complete, click **Done**.

Verifying the Installation

Verify that Replication Server has been successfully installed.

1. In a command window, change to the directory where you installed Replication Server.
2. Log in to Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
```

3. Verify the Replication Server version:

```
admin version  
go
```

4. Exit the **isql** session with the **quit** command.

Installing Replication Agent

Install Replication Agent using the GUI wizard.

Note: If there is not enough disk space in your default temporary directory, set the TMP Windows environment variable to a directory that has enough space. Include the full path to this directory.

1. Log in to the Replication Agent host machine using an operating system user account with authority to start, stop, and administer the Replication Agent instance (for example, the “sybase” user).
2. Close all nonessential applications, and minimize any open windows.
3. Download and extract the Replication Agent installation image from the SAP Service Marketplace (SMP).
4. Start the installation program. On Microsoft Windows platforms, the installation program should start automatically. If it does not, double-click the **setup.exe** file.
5. Click **Next**.
6. Select the geographic location, agree to the license agreement, then click **Next**.
7. Specify an installation directory. The default installation directory is the existing `%SYBASE%` or `c:\sybase`.

Click **Next** to accept the default installation directory, or:

- Click **Choose** to select an installation directory in the file browser. Click **Next**.
- Enter a directory name. Click **Next**.

8. Select the type of database the Replication Agent will connect to.

Click **Next**.

9. Select one of these options to enter the license:

- **Specify license keys** – browse to or specify the license file.
- **Use previously deployed license server** – use a previously deployed license server. Enter the host name of the machine where the license server is running and the port number if the port number you are using is not the default.
- **Continue installation without a license key** – install and use Replication Agent without a license for a grace period of 30 days. To continue using Replication Agent after the end of the grace period, obtain a valid license from the SAP Service Marketplace (SMP), and install it.

Note: Replication Agent is licensed statically when installed with RSHE. If you are installing Replication Agent with RSHE, locate the file that contains a valid SySAM license for the Replication Agent component in RSHE. Specify that license file here.

See the *SySAM Users Guide*.

Click **Next** until you see the SySAM Notification window.

10. On the SySAM Notification window, configure your server for e-mail notification. When configuration is enabled, you receive information about license management events that require attention. Select **Yes** and either accept the default values that are supplied, or enter values for:

- SMTP server host name
- SMTP server port number
- E-mail return address
- Recipients of the notification
- Message severity level of an event that triggers e-mail notification:
 - Informational
 - Warning
 - Error

If you choose not to have e-mail alerts or severity messages logged, select **No**.

Click **Next**.

11. Review the product features or components listed on the Preinstallation Summary window. Click **Install**.

12. Click **Done**.

Verifying the Installation

Verify that Replication Agent has been successfully installed.

1. In a command window, change to the directory where you installed Replication Agent.
2. Change to %SYBASE%\RAX-15_5\bin.
3. Obtain the Replication Agent version string:

```
ra.bat -v
```

If Replication Agent installs successfully, you see the SAP copyright and the Replication Agent version string.

Configuring Replication Components

Review the procedures for configuring each of the replication components in the replication system.

See also

- *Testing Replication* on page 78

Configuring ECDA Option for ODBC

Review the procedures for configuring ECDA Option for ODBC.

Configuring the Microsoft SQL Server ODBC Driver

Configure the ODBC driver to connect to the replicate database.

The ODBC driver is installed with the installation of Microsoft SQL Server.

Note: If you already have a data source configured for your replicate database, skip to "Configure and Start the DirectConnect Server."

1. Open the ODBC Driver Manager by navigating to **Control Panel > Administrative Tools** and double-clicking the **Data Sources (ODBC)** icon.
2. Click the **Drivers** tab, and verify that the Microsoft SQL Server driver is version 2000.86.1830.00. If it is an earlier version, go to the Microsoft Web site to download and install the correct version.
3. Click the **System DSN** tab, click **Add**, and select the Microsoft SQL Server driver. Click **Finish**.
4. In the Name field, enter a name that identifies your replicate data server and database, for example, `my_mssql_datasource_name`. Use this name later when you configure the ECDA access service. Select the replicate Microsoft SQL Server in the Server drop-down list. Click **Next**.
5. Be sure that Microsoft SQL Server authentication is selected, and the Connect to Microsoft SQL Server check box is checked. Enter a valid Microsoft SQL Server login ID and password, and click **Next**.
6. Select **Change the Default Database To**, and select the replicate database from the drop-down list. Click **Next**.
7. Click **Finish**.
8. Verify ODBC connectivity to the replicate database by clicking Test Data Source. If connectivity is not successful, verify that you have configured the ODBC driver correctly.
9. Click **OK**.
10. You should now see your Microsoft SQL Server data source in the list of System Data Sources. Click **OK**.

See also

- *DirectConnect Server Configuration and Startup* on page 59
- *Creating a Maintenance User for Replication in Microsoft SQL Server* on page 61

DirectConnect Server Configuration and Startup

Create, configure, and start a DirectConnect server.

Before you begin to create and configure the DirectConnect server, make sure that:

CHAPTER 4: Microsoft SQL Server Replication Setup

- The name for a valid Microsoft SQL Server account is available. This name is used as the administrator for the DirectConnect server.
- The number of the unused port to be used by the DirectConnect server is available.

See also

- *Creating a Maintenance User for Replication in Microsoft SQL Server* on page 61

Creating and Configuring a DirectConnect Server

Use the Create Server wizard to create a DirectConnect service.

1. Start the Create Server wizard by navigating to the %SYBASE%\DC-15_0\DCWizard directory and executing the DCWizard script. The Welcome Create Server wizard appears. Click **Next**.
2. Select **ECDA Option for ODBC**. Click **Next**.
3. Enter the server name for the new DirectConnect server and the port number that you want the server to listen on. Click **Next**.
4. Enter the ECDA access service name that you want to use for the replicate Microsoft SQL Server database, for example, `my_mssql_access_service`. Click **Next**.
5. Verify the DirectConnect server information. If correct, select **Create Server**. Otherwise, click **Back** to return to the previous window and provide the correct information.
6. Open the %SYBASE%\DC-15_0\servers\servername\cfg\dcany.cfg file, where *servername* is the name of the server you created.
7. In the file, locate the name of the Microsoft SQL Server access service that you created. Under that, set the **ConnectionSpec1** parameter to the name of the Microsoft SQL Server ODBC data source you created. For example:

```
[Service Library]
{Client Interaction}
SvcLibDescription=Access Service Library for ODBC
{Logging}
LogSvcLibStatistics=0
[my_mssql_access_service]
{ACS Required}
ConnectionSpec1=my_mssql_datasource_name
{Client Interaction}
EnableAtStartup=yes
TransactionMode=long
```

Note: TransactionMode=long indicates that the client session will manually issue the commit command for transactions.

8. Start the DirectConnect server:
 - a) Open a command window, and navigate to the DirectConnect installation directory. For example:

```
cd c:\sybase\DC-15_0
```

- b) Execute `DC_SYBASE.bat` to set the DirectConnect environment variables.
- c) Navigate to the DirectConnect `bin` directory:

```
cd bin
```

- d) Start the DirectConnect server. This also starts all of the configured access services:

```
DCStart -Sservername
```

where *servername* is the name of the DirectConnect server.

9. Click **Finish**.

See also

- *Verifying that you can Connect to Microsoft SQL Server* on page 61

Verifying that you can Connect to Microsoft SQL Server

Verify the connection to the replicate Microsoft SQL Server database.

1. Open a command window in the `%SYBASE%` directory of your ECDA installation.
2. Change to the `%SYBASE%\DC-15_0` directory.
3. Set the environment variables by running the `%SYBASE%\DC_SYBASE.bat` file.
4. Log in to the replicate database through the DirectConnect service:

```
isql -Uvaliduser -Ppassword -SDCService
```

where *validuser* and *password* are any valid user and password, and *DCService* is the DirectConnect service name.

5. Verify the connection to the replicate database:

```
select @@sqldbmsname
go
SQLDbmsName
-----
Microsoft SQL Server

select @@sqldbmsver
go
SQLDbmsVer
-----
09.00.1399
```

6. Exit the `isql` session with the `quit` command.

Creating a Maintenance User for Replication in Microsoft SQL Server

The maintenance user is a valid Microsoft SQL Server user that the Replication Server uses to apply commands to the replicate Microsoft SQL Server database.

1. Connect to the replicate database, and enter this command to create the maintenance user login:

CHAPTER 4: Microsoft SQL Server Replication Setup

```
CREATE LOGIN maintuser WITH PASSWORD='password',  
DEFAULT_DATABASE=replicate_db_name;
```

where *password* is the maintenance user login password, and *replicate_db_name* is the replicate database name.

2. At the replicate database, enter this command to create a maintenance user:

```
CREATE USER maintuser FOR LOGIN maintuser;
```

3. Grant database owner permission to the maintenance user:

```
EXEC sp_addrolemember 'db_owner', 'maintuser';
```

Configuring Replication Server

Review the procedures for configuring Replication Server for Microsoft SQL Server.

Configuring Replication Server for Replication to the Target Data Server

Create a Replication Server connection to the replicate data server.

1. Navigate to the %SYBASE%\RAX-15_5\scripts\mssql directory.
2. Make a copy of the `mssql_create_rs_standby_connection.sql` script named `my_mssql_create_rs_standby_connection.sql`.
3. Before executing the `my_mssql_create_rs_standby_connection.sql` script against your Replication Server, change *{rds}*.*{rdb}* to the name of the connection that Replication Server uses to connect to the DirectConnect access service, where:
 - *rds* is the DirectConnect access service name.
 - *rdb* is any valid identifier. SAP recommends that you use the Microsoft SQL Server replicate database name.
 - *maintuser* and *password* are the maintenance user and password created in ECDA Option for ODBC in step 1 of "Creating a maintenance user for replication." For example:

```
create connection to rds.rdb  
using profile rs_mss_to_msss;standard  
set username maintuser  
set password "password"  
go
```

Note: In the previous example, **password** is a Replication Server reserved word and therefore, must be enclosed in double quotes.

4. Create the connection to the replicate database:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i  
my_mssql_create_rs_standby_connection.sql
```

5. Log in to SAMPLE_RS, and verify the Replication Server connection to the standby database:

```
isql -Usa -Psa_pass -SSAMPLE_RS  
admin who  
go
```

You should see a message returned for the DSI connection *{rds}.{rdb}*. Verify that the status is “Awaiting Message” or “Awaiting Command.”

6. Exit the **isql** session with the **quit** command.

See also

- *Creating a Maintenance User for Replication in Microsoft SQL Server* on page 61

Configuring Replication Server for Replication from the Primary Database

Configure Replication Server for replication from the primary database by granting create object permission to the **rs_username** user.

The **rs_username** user, which is used for communication between Replication Agent and Replication Server, must have **create object** permission before you can use it to create a replication definition. You must grant this permission manually from Replication Server.

1. Log in to Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
```

If you have not set up the sample Replication Server instance, enter your Replication Server instance name in place of **SAMPLE_RS**.

2. Enter:

```
grant create object to SAMPLE_RS_ra
go
```

where *SAMPLE_RS_ra* is the value for the **rs_username** parameter you entered in the resource file.

Creating a Maintenance User on the Primary Connection

The maintenance user on the primary connection is a valid Microsoft SQL Server user whose activities are ignored if Replication Agent processes log records by the user. Replication Server requires one maintenance user to be defined for each connection.

1. Connect to the primary database, and enter this command to create the maintenance user login:

```
CREATE LOGIN maintuser WITH PASSWORD='password',
DEFAULT_DATABASE=primary_db_name;
```

where *password* is the maintenance user login password, and *primary_db_name* is the primary database name.

2. At the primary database, enter this command to create a maintenance user:

```
CREATE USER maintuser FOR LOGIN maintuser;
```

3. Grant database owner permission to the maintenance user:

```
EXEC sp_addrolemember 'db_owner', 'maintuser';
```

Creating a Replication Server Connection to the Primary Database

Create a Replication Server connection to the primary database.

1. Go to the %SYBASE%\RAX-15_5\scripts\sybase directory.
2. Make a copy of the `rs_create_test_primary_connection.sql` script named `my_mssql_rs_create_primary_connection.sql`.
3. Before executing the `my_mssql_rs_create_primary_connection.sql` script against your Replication Server, change `{rax.test}` to the name of the connection between Replication Agent and Replication Server, where:
 - `rax` is the name of the primary Microsoft SQL Server data server.
 - `test` is the name of the primary Microsoft SQL Server database.

For example, `NY.NYmss5`.

Note: Save these values to be used later for the `rs_source_ds` and `rs_source_db` parameters in the Replication Agent `mssql.rs` file.

4. Change `maint_user` and `maint_pwd` to the user ID and password of the Microsoft SQL Server maintenance user who must have permission to apply DML operations against all user tables to be replicated, where:
 - `maint_user` is the user ID of the Microsoft SQL Server maintenance user you created for the primary connection.
 - `maint_pwd` is the password of the Microsoft SQL Server maintenance user (`maint_user`).

Note: This user must be a valid user in the Microsoft SQL Server database, and it must be different from the user defined by the Replication Agent `pds_username` parameter, which is used for Replication Agent connections to the primary database.

This command creates a Replication Server connection to the primary database example:

```
create connection to NY.NYmss5
set error class rs_udb_error_class
set function string class rs_mss5_function_class
set username maint_user
set password change_on_install
with log transfer on, dsi_suspended
go
```

5. Execute the script in Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i
my_mssql_rs_create_primary_connection.sql
```

You see a message indicating that the Replication Server connection to the primary database has been created.

See also

- *Creating a Database Replication Definition* on page 65
- *Creating a Database Replication Subscription* on page 66

Granting create object Permission to the rs_username User

Grant create object permission to the user ID specified by the Replication Agent **rs_username** parameter.

The **rs_username** user, which is used for communication between Replication Agent and Replication Server, must have **create object** permission before it can be used to create a replication definition. You must grant this permission manually from Replication Server.

1. Log in to Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
```

2. Enter:

```
grant create object to SAMPLE_RS_ra
go
```

where SAMPLE_RS_ra is the value for the **rs_username** parameter you entered in the resource file.

3. Connect to the RSSD:

```
1>connect
2>go
```

4. Verify that the **rs_username** user has **create object** permission:

```
1>rs_helpuser
2>go
```

If **rs_username** has **create object** permission, the permission is displayed. For example:

```
SAMPLE_RS_ra connect source, create object
```

Creating a Database Replication Definition

Create a database replication definition.

1. Navigate to the %SYBASE%\RAX-15_5\scripts\sybase directory.
2. Make a copy of the rs_create_test_db_repdef.sql script named my_mssql_rs_create_test_db_repdef.sql.
3. Before executing the my_mssql_rs_create_test_db_repdef.sql script, change the value of *{pds}*.*{pdb}* to the name of the connection string you defined for the primary database, where:
 - *pds* is the name of the primary Microsoft SQL Server data server.
 - *pdb* is the name of the primary Microsoft SQL Server database.

For example, *NY.NYmss5*.

Note: Save these values to be used later for the **rs_source_ds** and **rs_source_db** parameters in the Replication Agent `oracle.rs` file. `mssql.rs` file.

This is a database replication definition example:

```
create database replication definition NY_repdef1
with primary at NY.NYmss5
go
```

4. Connect to the Replication Server and enter:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i
my_mssql_rs_create_test_db_repdef.sql
```

You see a message indicating that the database replication definition has been created.

See also

- *Creating a Database Replication Subscription* on page 66

Creating a Database Replication Subscription

Create a database replication subscription.

1. Make a copy of the `rs_create_test_db_sub.sql` script named `my_mssql_rs_create_test_db_sub.sql`.
2. Before executing the `my_mssql_rs_create_test_db_sub.sql` script, change `{pds}`, `{pdb}` and `{rds}`, `{rdb}` to the appropriate connection name, where:
 - `pds` is the name of the primary Microsoft SQL Server data server.
 - `pdb` is the name of the primary Microsoft SQL Server database.
 - `rds` is the DirectConnect access service name.
 - `rdb` is the name of the replicate database.

Note: Save the values of `pds` and `pdb` to be used later for the **rs_source_ds** and **rs_source_db** parameters in the Replication Agent `mssql.rs` file.

Here is an example of how to create a database replication subscription:

```
create subscription NY_sub1
for database replication definition NY_repdef1
with primary at NY.NYmss5
with replicate at my_mssql_access_service.my_repl_db
without materialization
go
```

3. Connect to the Replication Server and enter:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i
my_mssql_rs_create_test_db_sub.sql
```

You see a message indicating that the subscription is being created.

Configuring Replication Agent

Review the procedures for configuring Replication Server for Microsoft SQL Server.

Primary Microsoft SQL Server Configuration and Verification

Configure and verify the primary Microsoft SQL Server database for replication.

Verifying the Compatibility Level of Microsoft SQL Server

Verify that the database compatibility level is set to Microsoft SQL Server 2008, Microsoft SQL Server 2008 R2, or Microsoft SQL Server 2012.

To verify the compatibility level:

1. In the Microsoft SQL Server Management Studio, right-click your primary database and select **Properties**.
2. In Database Properties, click **Options**, and verify that the compatibility level is "SQL Server 2008 (100) or SQL Server 2012 (110)." If it is not, select that compatibility and click **OK**.

Disabling Microsoft SQL Server Replication

A Microsoft SQL Server publication cannot be created in the primary database used by Replication Agent for Microsoft SQL Server, and you cannot simultaneously use Microsoft replication and Replication Agent on the same Microsoft SQL Server database. If a Microsoft SQL Server publication already exists, remove the publication before using Replication Agent for Microsoft SQL Server.

Creating a Microsoft SQL Server User and Granting Permissions

Create a Microsoft SQL Server user named `ra_user`, and grant permissions to it. Log in to the primary Microsoft SQL Server as a system administrator, and run:

```
use master;
create login ra_user
  with password = 'my_pass';
use <primary_database>;
create user ra_user for login ra_user;
EXEC sp_addsrvrolemember 'ra_user', 'sysadmin';
```

where `my_pass` is a password that complies with the Microsoft SQL Server complexity policy.

Enabling Remote DAC

Enable remote DAC.

1. Log in to the server:

Microsoft SQL Server 2008	"C:\Program Files\Microsoft SQL Server\100\Tools\Binn\SQLCMD.EXE" -U <i>username</i> -P <i>password</i> -S <i>serverName</i>
Microsoft SQL Server 2012	"C:\Program Files\Microsoft SQL Server\110\Tools\Binn\SQLCMD.EXE" -U <i>username</i> -P <i>password</i> -S <i>serverName</i>

where *username*, *password*, and *serverName* are your user ID, password, and Microsoft SQL Server name.

2. Enable DAC to allow remote connections:

```
1> sp_configure 'remote admin connections', 1
2> go
```

3. Update the **remote admin connections** setting:

```
1> reconfigure
2> go
```

4. Verify the **remote admin connections** setting:

```
1> sp_configure 'remote admin connections'
2> go
```

The result returned should be:

name	minimum	maximum	config_value	run_value
remote admin connections	0	1	1	1

Obtaining the Microsoft SQL Server DAC Port Number

Obtain the Microsoft SQL Server DAC port number.

1. Use a text editor to open the ERRORLOG file. This file is located in the log directory of your Microsoft SQL Server. For example:

```
C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\LOG\ERRORLOG
```

2. Search for "Dedicated admin" to find an entry similar to this:

```
2007-11-09 13:40:02.40 Server Dedicated admin
connection support was established for listening
locally on port 1348.
```

3. Record the port number specified in this entry for use in a later step.

Verifying the Version of the Microsoft Filter Manager Library

Confirm that the Microsoft Filter Manager Library version is 5.1.2600.2978 or later, to make the primary transaction log files readable for Replication Agent.

To determine the version of the library:

1. Navigate to c:\windows\system32.

2. Right-click **ftlib.dll** and select **Properties**.

You see the **Properties** dialog box.

3. On the **Properties** dialog box, click the **Details** tab.
4. If the version is earlier than 5.1.2600.2978, go to the Microsoft Web site at <http://www.update.microsoft.com/windowsupdate/v6/vistadefault.aspx?ln=en-us>, and update your Windows system.

Making the Primary Transaction Log Files Readable for Replication Agent

Install and set up the `sybfilter` driver so that Replication Agent can read the primary transaction log files.

1. In Windows Explorer, navigate to the `sybfilter` driver installation directory. This directory is located at `%SYBASE%\RAX-15_5\system\, where <platform> is:

 - winx64 – for 64-bit Windows Server 2008, Windows Server 2008 R2, and Windows 7.
 - winx86 – for 32-bit Windows Server 2008, Windows Server 2008 R2, and Windows 7.
 - winvistax64 – for all other supported 64-bit Windows versions.`
2. To install the driver, right-click `sybfilter.inf`, and click **Install**.

Note: There can be only one installation of the `sybfilter` driver on a Windows machine. After the driver is installed, it works for all Replication Agent for Microsoft SQL Server instances running on the same machine.

3. Under any directory, create a configuration file to store all log file paths for primary databases. The configuration file must have a `.cfg` suffix. For example, under the directory `%SYBASE%\RAX-15_5\system\, create a file named LogPath.cfg.`
4. Add a system environment variable named `RACFGFilePath`, and set its value to the path of the configuration file.
 - a) Open the **Control Panel**, click **System**, click the **Advanced** tab, and choose **Environment Variables**.
 - b) Click **New** to add a new system variable.
 - c) Name the variable `RACFGFilePath`, and set its value to the location of your configuration file.
5. In Windows Explorer, navigate to `%SYBASE%\RAX-15_5\bin`, and right-click **sybfiltermgr.exe** and select **Run as administrator** to start the `sybfilter` driver management console.
6. To start the `sybfilter` driver, enter **start** at the management console.
7. Add the log file path to the `sybfilter` driver with the user manager or by modifying the configuration file:

CHAPTER 4: Microsoft SQL Server Replication Setup

- User manager – use the **add** command in the management console. The syntax for this command is:

```
add serverName dbName logFilePath
```

For example:

```
add myserverName dbName C:\Program Files\Microsoft SQL  
Server\MSSQL.1\MSSQL\Data\pdb2_log.ldf
```

The user manager refreshes all log paths in the `sybfilter` driver automatically after adding the log path into the configuration file.

- Configuration file – to add the log file path directly to the configuration file, open and manually edit the configuration file. This an example of log file path entries:

```
[myserver, pdb1]  
log_file_path=C:\Program Files\Microsoft SQL  
Server\MSSQL.1\MSSQL\Data\pdb11_log.ldf  
log_file_path=C:\Program Files\Microsoft SQL  
Server\MSSQL.1\MSSQL\Data\pdb12_log.ldf  
[myserver, pdb2]  
log_file_path=C:\Program Files\Microsoft SQL  
Server\MSSQL.1\MSSQL\Data\pdb2_log.ldf
```

After you have added the log file paths to the configuration file, use the **refresh** command in the management console.

8. If you added a log file, restart Microsoft SQL Server to make the log file readable.
9. At the management console, verify that log files are readable. If some log files are unreadable, make sure the files have been created and that Microsoft SQL Server has been restarted, if necessary.

See the *Replication Agent Primary Database Guide*.

Stopping the Analysis Service

Terminate the SQL Server Analysis Service if it is running.

1. From the Windows Control Panel, select **Administrative Tools > Services**.
2. Find the service named SQL Server Analysis Service (*SERVER*), where *SERVER* is the name of your Microsoft SQL Server data server.
3. Stop this service.

Enabling TCP/IP

Enable TCP/IP.

1. Go to **Microsoft SQL Server > Configuration Tools**, and choose **SQL Server Configuration Manager > SQL Server Network Configuration > Protocols for [SQLInstanceName]**, where *SQLInstanceName* is your SQL Server instance.
2. Right-click **TCP/IP**, and choose **Enable**.

Installing the Microsoft SQL Server JDBC Driver

Install the Microsoft SQL Server JDBC driver.

1. Go to the Microsoft download site at <http://www.microsoft.com/downloads> and search for:

Microsoft JDBC Driver 4.0
2. Click the link for the 4.0 driver, and follow the instructions to download and install the driver.
3. Open **Control Panel > System**, click the **Advanced** tab, and click **Environment Variables**.
4. In the System Variables window, if the CLASSPATH variable is not listed, click **New**. Otherwise, select it, and click **Edit**.
5. Enter the full path of the JDBC driver named `sqljdbc4.jar`, using a semicolon (;) to separate it from any other drivers.
6. Click **OK** three times.

Configuring and Validating the Replication Agent Resource File

A single installation of the Replication Agent can support replication from multiple databases. However, one Replication Agent instance is needed for each Microsoft SQL Server database that is to be replicated.

1. Locate the resource file template.

The majority of configuration values required to create and initialize a Replication Agent can be recorded and stored in a resource file. Using a resource file provides a means to record or retain the configuration information for a Replication Agent instance, allowing an instance to be removed and re-created.

The Microsoft SQL Server resource file template is at `%SYBASE%\RAX-15_5\init\mssql.rs`.

2. Create an instance resource file:

Copy the resource file template `%SYBASE%\RAX-15_5\init\mssql.rs` to another file, for example, `myra.rs`.

Change the values assigned to properties in the resource file so they match the values required for your environment. The majority of these values are host, port, user, and password information for the Microsoft SQL Server, Replication Server, and RSSD connections.

- Set the value of **rs_source_ds** to the value you chose for *rax* and *pds* in previous procedures.
- Set the value of **rs_source_db** to the value you chose for *test* and *pdb* in previous procedures.

CHAPTER 4: Microsoft SQL Server Replication Setup

- When you create a Replication Agent instance with a resource file:
 - The **asa_password** configuration parameter value must not contain single quotes, double quotes, or a semicolon.
 - The **pds_username** and **pds_password** configuration parameter values must not contain single or double quotes if the **create_pds_username** parameter is set to yes.

The resource file is self-documenting, with each parameter described. See the *Replication Agent Administration Guide*.

Warning! The combined values of **rs_source_ds** and **rs_source_db** must match the *{pds.pdb}* values of the Replication Server primary connection name.

Table 5. Resource File Parameters

Parameter	Description	Example values
instance name	Any valid name.	myra
admin_port	Port number that Replication Agent uses.	20500 (if in use, select a different port number)
pds_server_name	Name of the primary Microsoft SQL Server.	TEAMSTER
pds_port_number	Port number for the primary Microsoft SQL Server.	1433
pds_dac_port_number	Microsoft SQL Server Dedicated Administration connection port number.	1348
pds_database_name	Name of the primary database.	test_db
pds_username	User ID that Replication Agent uses to access the primary data server.	ra_user
pds_password	Password for pds_username .	sybase
rs_host_name	Machine where Replication Server is installed.	teamster
rs_port_number	Port where Replication Server is installed.	11752
rs_username	Replication Server user with connect source and create object permissions.	SAMPLE_RS_ra
rs_password	Password for rs_username .	SAMPLE_RS_ra_ps
rs_source_ds	Valid name representing data server of primary Microsoft SQL Server database.	NY
rs_source_db	Valid name representing primary Microsoft SQL Server database.	NYmss5

Parameter	Description	Example values
rs_charset	Character set that Replication Server is using. Note: The value defined for the rs_charset configuration parameter must match the RS_charset value in the Replication Server configuration file, %SYBASE%\REP-15_5\install\<server>.cfg.	cp850
rssd_host_name	Machine where RSSD resides.	teamster
rssd_port_number	Port number where RSSD resides.	11751
rssd_database_name	Database name for RSSD.	SAMPLE_RS_ERSSD
rssd_username	Valid user for RSSD.	SAMPLE_RS_RSSD_maint
rssd_password	Password for rssd_username .	SAM- PLE_RS_RSSD_maint_ps
start_instance	Automatically start the instance that was created.	yes
initialize_instance	Automatically initialize the new Replication Agent instance.	yes

3. Be sure your CLASSPATH environment variable points to the Microsoft SQL Server JDBC driver, and use the Replication Agent **ra_admin** utility to validate the settings in the new instance resource file using the **-vr** parameter. For example:

```
%SYBASE%\RAX-15_5\bin\ra_admin -vr myra.rs
```

Validation results are returned as either:

- Response-file processing completed
- Or
- Response-file processing completed with errors

If any validation fails, the **ra_admin** utility returns an error message and information about the failure. You can repeat the validation process as many times as necessary until it executes without error. No entities are changed or created.

See also

- *Initialization* on page 74
- *Subsequent Initialization* on page 76
- *Verifying the Replication Agent Replicating State* on page 77

Initialization

To initialize the primary data server, Replication Agent for Microsoft SQL Server installs objects at the data-server level and at the database level.

Data-server level modifications are required only once.

If you have previously initialized the primary data server and a Replication Agent instance and want to initialize another Replication Agent instance, see *Subsequent Initialization*.

See also

- *Subsequent Initialization* on page 76
- *Verifying the Replication Agent Replicating State* on page 77

Configuring the Primary Data Server and a Replication Agent Instance for the First Time

Configure the primary data server and a Replication Agent instance for the first time.

1. Stop the Microsoft SQL Server service.
 - a) In **Control Panel > Administrative Tools > Services**, find the service named Microsoft SQL Server (*SERVER*), where *SERVER* is the name of your Microsoft SQL Server data server. For example:

```
Microsoft SQL Server (TEAMSTER)
```

- b) Stop the service.

2. Open a command window, and restart Microsoft SQL Server in single-user mode:

```
"C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\Binn  
\sqlservr.exe" -m -s instanceName
```

where *instanceName* is the name of the Microsoft SQL Server instance.

3. Execute the resource file to create the Replication Agent instance. In your resource file, if you set **start_instance** to **yes**, this step also starts the Replication Agent instance. In your resource file, if you also configured **initialize_instance** to **yes**, this step also initializes the Microsoft SQL Server and the Replication Agent instance.

After validating the resource file, create the Replication Agent instance:

```
%SYBASE%\RAX-15_5\bin\ra_admin -r myra.rs
```

Execution results are returned as either:

- Response-file processing completed
- or:
- Response-file processing completed with errors

See the *Replication Agent Administration Guide*.

4. If your resource file set **start_instance** to **no**, change to the SYBASE directory and start the Replication Agent instance:


```
cd %SYBASE%\RAX-15_5\myra
```

Execute the RUN file, for example:

```
RUN_myra
```

5. Use **dsedit** to update the Replication Server `sql.ini` to include an entry for the Replication Agent location.
6. Verify the connection to Replication Agent.
 - a) Open a command window in the `%SYBASE%` directory of your Replication Agent installation.
 - b) Set the environment variables by executing the `SYBASE.bat` file.
 - c) Log in to Replication Agent:


```
isql -Usa -Psa_pass -Smyra
```
7. If your resource file has **initialize_instance** set to **no**, initialize the primary data server and the Replication Agent instance:
 - a) Open a command window, and log in to the Replication Agent instance.
 - b) Issue this command to create server administration objects:


```
server_admin init
```
 - c) Issue this command to shutdown Replication Agent:


```
shutdown
```
8. Stop the Microsoft SQL Server in single-user mode:
 - a) Log in to the server:


```
"C:\Program Files\Microsoft SQL Server\90\Tools\Binn\SQLCMD.EXE" -U username -P password -S serverName
```

where *username*, *password*, and *serverName* are your user ID, password, and Microsoft SQL Server name.
 - b) Issue the **shutdown** command.
9. Restart Microsoft SQL Server in multiuser mode (normal start):
 - a) In **Control Panel > Administrative Tools > Services**, find the service named Microsoft SQL Server (*SERVER*), where *SERVER* is the name of your Microsoft SQL Server data server. For example:


```
Microsoft SQL Server (TEAMSTER)
```
 - b) Start the service.
10. Restart Replication Agent.
 - a) Change to the SYBASE directory and start the Replication Agent instance:


```
cd %SYBASE%\RAX-15_5\myra
```
 - b) Execute the RUN file, for example:


```
RUN_myra
```
11. Initialize Replication Agent.

CHAPTER 4: Microsoft SQL Server Replication Setup

- a) Log in to Replication Agent using isql:

```
isql -Usa -P -Smyra
```

where *myra* is the name of Replication Agent.

- b) Issue:

```
ra_admin init
```

The Replication Agent instance is now running, and the primary data server and the Replication Agent instance have been initialized.

Subsequent Initialization

Create another Replication Agent instance, after you have already initialized the primary data server.

See also

- *Verifying the Replication Agent Replicating State* on page 77

Creating and Initializing a Replication Agent Instance

Execute the resource file to create the Replication Agent instance.

1. After validating the resource file, create the Replication Agent instance:

```
%SYBASE%\RAX-15_5\bin\ra_admin -r myra.rs
```

Execution results are returned as either:

- Response-file processing completed
- or:
- Response-file processing completed with errors

See the *Replication Agent Administration Guide*.

2. If your resource file has **initialize_instance** set to **no**, start the Replication Agent instance.

- a) Change to the directory containing your Replication Agent instance RUN file, for example:

- b) Execute the RUN file, for example:

```
RUN_myra
```

3. Use **dsedit** to update the Replication Server `sql.ini` to include an entry for the Replication Agent location.

4. Verify the connection to Replication Agent.

- a) Open a command window in the `%SYBASE%` directory of your Replication Agent installation.
- b) Set the environment variables by executing the `SYBASE.bat` file.
- c) Log in to Replication Agent:

```
isql -Usa -Psa_pass -Smyra
```

5. If your resource file has **initialize_instance** set to **no**, open a new command window, log in to the Replication Agent instance, and initialize the Replication Agent instance and the primary Microsoft SQL Server database using this Replication Agent command:

```
ra_admin init
```

Your Replication Agent instance is now running and has been initialized.

Verifying the Replication Agent Replicating State

Verify the replicating state of Replication Agent.

1. Enter:

```
resume
go
```

If the Replication Agent successfully transfers to a replicating state, you see:

```
State          Action
-----
REPLICATING Ready to replicate data.
```

The Replication Agent goes to the REPLICATION DOWN state if an error occurs:

```
isql -Usa -Psa_pass -SSAMPLE_RS
1> resume
2> go
Msg 32000, Level 20, State 0:
Server 'myra', Procedure 'resume', Line 1:
Command <resume> failed - Desired state <REPLICATE>
could not be achieved. Current state: <REPLICATION
DOWN>
```

2. The **ra_status** command returns the state of the Replication Agent. Verify that the Replication Agent remains in replicating state, even after the resume command executes successfully.

To detect an error that occurred after replication start-up, execute:

```
ra_status
go
```

If the Replication Agent is in replicating state, **ra_status** returns:

```
State          Action
-----
REPLICATING Ready to replicate data.
```

Replication Agent goes to the REPLICATION DOWN state if an error occurs.

3. Validate that both primary and replicate connections are active:

```
isql -Usa -Psa_pass -SSAMPLE_RS
admin who
go
```

Note: Be aware that:

CHAPTER 4: Microsoft SQL Server Replication Setup

- The DSI connection for the primary database connection is usually suspended because you are not replicating data back to the primary database.
- The Replication Agent connection, if established for the replicate database connection, is usually down, because you are not replicating data from the replicate database.

Do not proceed until **admin who** returns status for threads similar to this:

```
Spid Name          State          Info
-----
13  DSI EXEC        Awaiting Command 101(1) SAMPLE_RS_ERSSD.
SAMPLE_RS_ERSSD
9   DSI            Awaiting Message 101 SAMPLE_RS_ERSSD.
SAMPLE_RS_ERSSD
8   SQM            Awaiting Message 101:0 SAMPLE_RS_ERSSD.
SAMPLE_RS_ERSSD
8   SQM            Awaiting Message 101:0 SAMPLE_RS_ERSSD.
SAMPLE_RS_ERSSD
54  DSI EXEC        Awaiting Command 102(1)
my_mssql_access_service.rdb
53  DSI            Awaiting Message 102
my_mssql_access_service.rdb
17  SQM            Awaiting Message 102:0
my_mssql_access_service.rdb DSI
EXEC Suspended      103(1) pds.pdb
DSI Suspended        103 pds.pdb
24  DIST          Awaiting Wakeup  103 pds.pdb
25  SQT           Awaiting Wakeup  103:1 DIST pds.pdb
23  SQM           Awaiting Message 103:1 pds.pdb
22  SQM           Awaiting Message 103:0 pds.pdb
62  REP AGENT     Awaiting Command pds.pdb
```

Testing Replication

When you finish setting up the replication system, test that replication works as intended.

1. Connect to the primary Microsoft SQL Server database as a regular user, rather than as the maintenance user. Make sure the regular user also exists in the replicate database.
2. Create a test table to replicate in the primary database:

```
create table T1 (a int, b char(10))
go
```

3. Grant permissions to any new or existing object to be replicated in the primary database:

```
grant all on T1 to public;
```

4. Connect to the replicate Microsoft SQL Server database as a regular user, rather than as the maintenance user.

5. Create a test table in the replicate database:

```
create table T1 (a int, b char(10))
go
```

6. Grant permissions to any new or existing object to be replicated in the replicate database so that the Replication Server maintenance user can update this table:

```
grant all on T1 to public
go
```

7. Connect to Replication Agent, and mark the table:

```
pdb_setreptable T1, mark
```

8. After the table is marked and the Replication Agent is in a replicating state, insert test data into the test table and commit it.

By default, the work performed by the maintenance user of the primary connection is not replicated. The user ID used to insert transactions cannot be the same as the maintenance user defined in the primary connection.

Replication Agent applies only committed transactions to a replicate database.

```
insert into T1 values ( 42, 'foo' )
go
```

9. Use your preferred query tool to examine the replicate site for results and compare the contents of your test table from both the primary and replicate sites.

See also

- *Configuring Replication Server for Replication from the Primary Database* on page 63

Resetting the Primary Microsoft SQL Server Database for Replication

In a test environment, you may occasionally need to reset the replication environment.

1. Protect your new environment from old log information by archiving all current redo log data. In the Replication Agent, run:

```
ra_locator move_trunct
```

2. Reinitialize Replication Agent, which refreshes the Replication Agent repository rather than overwriting it:

```
ra_admin refresh
```

Note: If you prefer to delete and replace all the information in the Replication Agent repository, issue **ra_admin deinit, force** followed by **ra_admin init** (without the **force** option).

3. Reset the locator stored in Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
connect
go
rs_zeroltm pds, pdb
go
exit
go
```

You see:

CHAPTER 4: Microsoft SQL Server Replication Setup

```
Locator has been reset to zero.  
(return status = 0)
```

Implement a sample replication system for DB2 UDB.

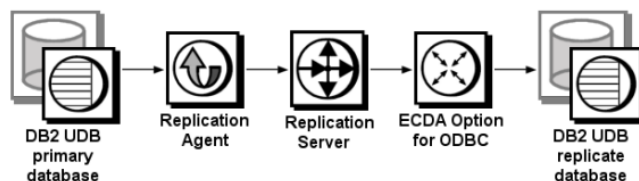
Note: Use the tasks in the *Replication Server Options Quick Start Guide* to set up replication only for proof-of-concept or testing, not for production.

System Architecture

There are multiple components required to implement a replication system for DB2 UDB.

- A primary DB2 UDB database
- A replicate DB2 UDB database
- An SAP Replication Server
- The SAP Replication Server Options components:
 - Replication Agent
 - Enterprise Connect™ Data Access (ECDA) Option for ODBC

Figure 3: SAP replication system for DB2 UDB



Prerequisites

Review the prerequisites before you begin setting up replication for DB2 UDB.

Make sure:

- You have an understanding of SAP replication and are familiar with DB2 UDB.
- You have valid licenses for Replication Server and Replication Server Options.
- DB2 UDB 9.5, 9.7, 10.1, or 10.5 is installed and configured to serve as the source (primary) database and as the target (replicate) database.
- TCP/IP connectivity is available.
- Any operating system patches required for Java have been installed.

- The DB2 UDB `java_heap_sz` configuration parameter is set to at least 2048.

Do not use Replication Server reserved words for object names or connection names. A complete list of reserved words is in the *Replication Server Reference Manual*.

Planning Your Installation

Before installing the replication components, verify the system requirements, identify the installation directories, and obtain the installation software.

System Requirements

Make sure the target computer on which you are installing the replication components meets the minimum memory and disk space requirements.

For Windows:

Components	Memory (RAM)	Disk space
Replication Server	512MB	380MB
ECDA	512MB	440MB
Replication Agent	512MB	300MB

For UNIX:

Components	Memory (RAM)	Disk space
Replication Server	512MB	520MB
ECDA	256MB	700MB
Replication Agent	512MB	300MB

For Linux:

Components	Memory (RAM)	Disk space
Replication Server	512MB	400MB
ECDA	256MB	590MB
Replication Agent	512MB	300MB

To improve performance by avoiding multiple network hops, install Replication Server, ECDA for ODBC, and the target database on the same machine.

See the *Replication Server Options Release Bulletin*.

Identify the Installation Directory

Determine the directory in which to install the replication components.

Replication Agent, Replication Server, and ECDA are installed in the same base directory, which is identified by the SYBASE environment variable. Select a path on the host drive to be the recipient of the installation and configuration activities. Replication Agent must be installed on the same host as either the DB2 UDB server or the DB2 UDB Administration Client.

Replication Agent Installation Directory

Replication Agent versions 15.5 and later use the RAX-15_5 directory name. If you have an existing Replication Agent installation of version 15.5 or later and do not want to overwrite it, install the latest version of Replication Agent in a different location.

Accessing and Downloading Products

Download replication components from SAP Service Marketplace (SMP) or the Sybase Product Download Center (SPDC).

- For SMP:
 1. Go to the SAP Web site <https://websmp204.sap-ag.de/patches> (login required).
 2. Select **SAP Software Download Center > Installations and Upgrades > A-Z Index** and click **R** for SAP Replication Server products.
 3. Click the Replication Server software you need by version and platform, and follow the instructions.
- For SPDC:

Note: If you are installing Replication Server Options (RSO), this task requires you to perform several steps twice, as you are downloading Replication Server and other products.

1. Go to <https://sybase.subscribenet.com/control/sybs/login>, and enter your login ID and password.
2. Select **Replication Server**.
3. From the high-level Product Information page, select **Replication Server** again.
4. If you are installing Replication Server Heterogeneous Edition (RSHE), skip to Step 7. If you are installing RSO, select the Replication Server software you need by version and platform.
The Software Terms and Conditions page appears, listing countries and regions.
5. Either:
 - Select **I Agree**, go to the Product Download Page, and continue with step 6, or
 - Select your country or region to read the terms.
The software licence agreement for the software appears. Read the terms and either:

- Select **I Agree** to accept the terms, go to the Product Download Page, and continue with step 6, or
 - Select **Cancel** to return to the high-level Product Information page.
6. The Product Download Page lists the software and documentation for the product. Select **FTP Download** or **Advanced Download Options** for the Replication Server software to download and save the compressed files to your local drive.

Note: For each product, the documentation download contains installation-related documents, and the rest of the documentation set can be accessed online.

7. Return to the detailed Product Information page that you accessed in step 3 and select:
 - The RSO software by primary database, version, and platform, or
 - The RSHE software
8. Navigate the Software Terms and Conditions pages as you did in step 5.
9. Uncompress and extract all the installation images to your local drive.

Installing Replication Components

Review the procedures for installing the replication components.

Install all replication components on the same host where a DB2 UDB data server has already been installed and is running with both the primary and replicate databases. This simplifies the quick-start process and avoids the step of separately installing DB2 UDB clients for Replication Agent and for ECDA Option for ODBC.

See also

- *Configuring Replication Components* on page 94

Installing ECDA Option for ODBC

Install ECDA Option for ODBC using the setup program.

Prerequisites

Install ECDA on the same server as the replicate database to eliminate a network hop and to improve performance.

Task

Note: To install using the console mode or a response file, see the *ECDA Installation Guide for Linux and UNIX* or the *ECDA Installation Guide for Microsoft Windows*.

1. Verify that the drive on which you will install the products has enough free disk space for the products being installed, and at least 10MB extra disk space for the installation program. The program frees this extra space after the installation is complete.

2. Insert the ECDA CD-ROM into the CD-ROM drive.

- On Microsoft Windows, the installation program should start automatically. If it does not, select **Start > Run**, and enter:

```
x:\setup.exe
```

where *x*: is your CD drive.

- On Linux or UNIX, if auto-mount is not available on your machine, follow local procedures for mounting the CD-ROM, and enter:

```
./setup.bin
```

The **Welcome** window appears.

3. Click **Next**. You may see the following error message:

```
Error writing file = There may not be enough temporary disk space. Try using -is:tempdir to use a temporary directory on a partition with more disk space.
```

If so, set the temporary directory to another directory that has more disk space by entering the following at the command line:

- (Windows)

```
x:\setup.exe -is:tempdir <directory_name>
```

- (UNIX or Linux)

```
./setup.bin -is:tempdir <directory_name>
```

where *<directory_name>* is the name of the temporary directory to which the installation program will write its temporary files.

4. Read the License and Copyright Agreement. Using the drop-down list at the top of the window, select the geographic location where you are installing ECDA to display the agreement appropriate to your region. Select **I agree**, and click **Next**.

You must agree to the license and copyright agreement before you can continue.

5. In the installation directory window, click **Next** to accept the default directory for the installation, or enter a different directory where you want to install ECDA. One of the following occurs:

- If the installation directory you chose does not exist, the installation program prompts:
The directory does not exist. Do you want to create it?

Click **Yes**.

- If the installation directory exists and contains files, the software prompts:

```
Warning: You have chosen to install into an existing directory. If you proceed with this installation, any older versions of the products you choose to install that are detected in this directory will be replaced.
```

Note: You will not see this message if the directory exists but is empty.

If you are prompted to overwrite any files or DDL, select **Yes** only if the version of the new files or DDL is later than the one you are attempting to overwrite.

Note: In certain cases when ECDA is installed with other SAP products, you may see warnings about overwriting newer versions of files. In these cases, simply instruct the installer to overwrite these files and proceed with the installation by selecting **Yes to All**.

6. Select one of the two installation types:

- Custom, which allows you to select the ECDA options and components to install.
- Full, which installs all ECDA products and components from the CD.

Click **Next**.

7. Based on your selection:

- If you selected Custom, the next window displays all the ECDA options and all the components “checked” or selected. You must “uncheck” or deselect the components that you do not want to install.

Note: Be aware that if you deselect a component that is required, it is automatically installed if it is needed to run other selected components.

After you have selected either the ECDA Option for ODBC or ECDA Option for Oracle and have selected the desired components, click **Next**.

- If you selected Full install, the next window displays all the ECDA options and all the components “checked” or selected. Click **Next**.

Before proceeding to the next window, the installation program verifies the selections, and checks for dependencies and available disk space. The **Product Summary** window displays every component that is to be installed and the total disk space required for all the selected components.

Note: If the target directory does not have enough free space, the space-required and the space-available information is displayed. If you click **Next** without sufficient hard disk space, an error occurs that stops the installation.

8. Verify that you have selected the correct type of installation, and that you have enough disk space to complete the process. Click **Next** to continue the installation.

The installation program unloads all the components from the CD and displays a progress indicator. When the installation is complete, a message appears indicating that the installation program wizard has completed installing your product.

9. Click **Next**. The SySAM License Server window opens and displays this prompt:

Will licenses be obtained from the License Server?

- Select **Yes** if you have a pre-existing SySAM network license server installed or will install a new license server.

Enter the host name and the port number of the license server.

- Select **No** if you do not have a SySAM license server installed and are going to use a local license file. A message appears telling you to download the license file.

If you select **Yes** and the license server is not found, or if you select **No** and the local license file is not installed on your local machine, this message appears:

```
Installer can't check out a license. Do you want to continue
without a license?
```

- 10.** Click **Yes** to complete the installation. If you select **No**, this message appears:

```
Please remember to download and install the license file
after this installation.
```

Click **OK**.

The SySAM Notification window asks you to configure your server for e-mail notification. When configuration is enabled, you will receive information about license management events requiring attention.

- 11.** Enter the following:

- SMTP server host name
- SMTP server port number
- E-mail Return Address
- Recipient e-mail addresses
- Message severity that triggers e-mail messages

Click **Next**.

- 12.** A window displays a message indicating that the installation was successful and advising you to check for software updates. Click **Finish**.

Installing Replication Server

Install Replication Server using the setup program.

Prerequisites

- Allocate a disk partition of at least 20MB for each Replication Server you are installing. You can add more partitions later, if necessary. Make sure each partition is available and has write permissions.
- Allocate the entire partition to the Replication Server. If you allocate only a portion of the partition to Replication Server, you cannot use the remainder for any other purpose.
- The sample Replication Server instance SAMPLE_RS used in these procedures must be created when you install Replication Server. If you have already installed Replication Server, see the instructions for how to set up a Replication Server instance after installation in *Replication Server ASE-to-ASE Replication Quick Start Guide > SAMPLE_RS*.

Task

1. Verify that the drive on which you install the Replication Server has enough disk space for the components being installed, and at least 100MB of extra disk space for the installation program.
2. Download and extract the Replication Server installation image from the Sybase Product Download Center (SPDC) or the SAP® Service Marketplace (SMP).
3. Start the installation.

- On Windows, launch the **setup** program.
If the installer does not start automatically, double-click `setup.exe` or select **Start > Run** and enter the following:

```
setup.exe
```

If there is not enough disk space in the temporary disk space directory, set the environment variable `TMP` to *directory_name* before running it again, where *directory_name* is the full path to and name of the temporary directory to which the installation program writes the temporary installation files.

- (UNIX or Linux) If you downloaded the product from SPDC, go to the directory where you extracted the installation image and start the installer:

```
./setup.bin
```

where `setup.bin` is the executable file name for installing Replication Server.

If there is not enough disk space in the temporary disk space directory, set the environment variable `IATEMPDIR` to *tmp_dir* before running the installer again, where *tmp_dir* is the full path to the temporary directory to which the installation program writes the temporary installation files.

4. In the Introduction window, click **Next**.
5. Specify where to install Replication Server:

Option	Description
Click Choose .	To select an installation directory. A window appears, and from that window browse for and select the directory.
Enter a new directory path.	To create a new directory.
Click Restore Default Folder .	To restore to the default directory if you do not want to use the directory that you have entered.

- If the directory you chose does not exist, click **Yes** to create it.

- If it does exist, and already contains a Replication Server installation, you are warned that you will overwrite the older version. Click **Next**.

6. Select the type of installation:

Option	Description
Typical	Installs the default components. This is recommended for most users.
Full	Installs every Replication Server component, including all the supported language modules.
Custom	Lets you select the components to install. Some components are automatically installed if they are required to run your selected components.

Click **Next**.

7. Select the geographic location, agree to the license agreement, then click **Next**.
8. On the SySAM License Server window:

Option	Choose
Specify license keys	<p>Either:</p> <ul style="list-style-type: none"> • Click Browse to select the license file. • Use Shift+Click or Ctrl+Click to select multiple license files. The license pane displays the license information. • Copy and paste the license information directly in the license pane. <p>Click Next.</p> <p>If you specify a served license key, the installer prompts you to install a new SySAM license server. Select:</p> <ul style="list-style-type: none"> • Next – to install the new SySAM license server. Follow the installation prompts. • Previous – and select Use previously deployed license server if you have an existing SySAM license server on the same host.
Use previously deployed license server	<p>Enter the:</p> <ul style="list-style-type: none"> • Host name of the machine where the license server is running • Port number if the port number you are using is not the default <p>Click Next.</p>

Option	Choose
<p>Continue installation without a license key</p>	<p>If you do not have licenses for any of the Replication Server components, select this option, then click Next to proceed.</p> <p>The installer allows you to install and use Replication Server components without a license for a grace period of 30 days. To continue using these components after the end of the grace period, obtain valid licenses, and install these licenses using the Replication Server License Installer.</p>

Use **sysadmin lmconfig** to configure and show license management-related information in Replication Server. See *Replication Server Reference Manual > Replication Server Commands > sysadmin lmconfig*.

9. Set SySAM e-mail configuration if necessary. Click **Next**.
10. The installation summary window displays the selections you have made. Review the information, and click **Install**.
11. In the Start Sample Replication Server window, select:

Option	Description
<p>Yes</p>	<p>Note: You cannot start the sample Replication Server on Linux on POWER because the ERSSD requires SAP® SQL Anywhere® Server, which is not available on Linux on POWER.</p> <p>To configure and start a sample Replication Server. The installer displays the configuration information for the sample Replication Server. Record this information.</p> <p>You must enter and confirm a password from 6 to 15 characters in length.</p>
<p>No</p>	<p>To manually configure a full-featured Replication Server and start a sample Replication Server after installation. The installer prompts you to either create a sample Replication Server directory or to proceed with the installation. Select either:</p> <ul style="list-style-type: none"> • Yes – to create a sample Replication Server directory without starting the sample Replication Server, or, • No – to continue with the installation.

Click **Next**.

12. When the installation is complete, click **Done**.

Verifying the Installation

If you have configured and started a sample Replication Server, verify that Replication Server has been successfully installed.

Prerequisites

Task

1. In a command window, change to the directory where you installed Replication Server.
2. Set the environment variables by sourcing the `SYBASE.csh` file (UNIX or Linux) or by executing the `SYBASE.bat` file (Windows).
3. Log in to Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
```

4. Verify the Replication Server version:

```
admin version
go
```

5. Exit the `isql` session with the `quit` command.

Installing Replication Agent

Install Replication Agent using the GUI wizard.

Note: If there is not enough disk space in your default temporary directory, set the `IATEMPDIR` (Linux or UNIX) or `TMP` (Windows) environment variable to a directory that has enough space. Include the full path to this directory.

1. Log in to the Replication Agent host machine using an operating system user account with authority to start, stop, and administer the Replication Agent instance (for example, the “sybase” user).
2. Close all nonessential applications, and minimize any open windows.
3. Download and extract the Replication Agent installation image from the SAP Service Marketplace (SMP).
4. Start the installation program:
 - On Microsoft Windows platforms, the installation program should start automatically. If it does not, start the installation program from Microsoft Windows Explorer by double-clicking the `setup.exe` file.
 - On UNIX platforms, enter:


```
./setup.bin
```
5. Click **Next**.

Note: You can click **Cancel** to stop the installation at any point.

6. Select the geographic location, agree to the license agreement, then click **Next**.
7. Specify an installation directory. The default installation directory is:
 - Existing `%SYBASE%` or `c:\sybase` on Microsoft Windows platforms
 - Existing `$SYBASE` or `/opt/sybase` on UNIX platforms

Click **Next** to accept the default installation directory, or:

- Click **Choose** to select an installation directory in the file browser. Click **Next**.
- Enter a directory name. Click **Next**.

If you continue and the older products were not installed with the current version of Replication Agent, the installation program overwrites any common files.

If the directory name does not exist, click **Yes** when prompted to create it:

```
The directory does not exist.  
Do you want to create it?
```

If the directory you specified exists, you see:

```
Warning: You have chosen to install into an existing  
directory. If you proceed with this installation,  
any older versions of the products you choose to  
install that are detected in this directory will be  
replaced.
```

Note: On Microsoft Windows platforms, if you are prompted to overwrite a DLL, click **Yes** only if the version of the new DLL is newer than the one the installation program is attempting to overwrite.

8. Select the type of database the Replication Agent will connect to.

Click **Next**.

9. Select one of these options to enter the license:
 - **Specify license keys** – browse to or specify the license file.
 - **Use previously deployed license server** – use a previously deployed license server. Enter the host name of the machine where the license server is running and the port number if the port number you are using is not the default.
 - **Continue installation without a license key** – install and use Replication Agent without a license for a grace period of 30 days. To continue using Replication Agent after the end of the grace period, obtain a valid license from the SAP Service Marketplace (SMP), and install it.

Note: Replication Agent is licensed statically when installed with RSHE. If you are installing Replication Agent with RSHE, locate the file that contains a valid SySAM license for the Replication Agent component in RSHE. Specify that license file here.

See the *SySAM Users Guide*.

Click **Next** until you see the SySAM Notification window.

10. On the SySAM Notification window, configure your server for e-mail notification. When configuration is enabled, you receive information about license management events that require attention. Select **Yes** and either accept the default values that are supplied, or enter values for:

- SMTP server host name
- SMTP server port number
- E-mail return address
- Recipients of the notification
- Message severity level of an event that triggers e-mail notification:
 - Informational
 - Warning
 - Error

If you choose not to have e-mail alerts or severity messages logged, select **No**.

Click **Next**.

11. Review the product features or components listed on the Preinstallation Summary window. Click **Install**.

The installation program installs the components in the installation directory you specified, and displays an installation progress indicator.

If errors occur during the installation, the installation program displays error messages. Exit the installation program wizard to correct the cause of the error, then restart the installation program.

If the software is installed successfully, you see a window confirming the successful installation.

12. Click **Done** to complete the installation and close the installation program.

Verifying the Installation

Verify that Replication Agent has been successfully installed.

1. In a command window, change to the directory where you installed Replication Agent.
2. Set the environment variables by sourcing the `SYBASE.csh` file (UNIX or Linux) or by executing the `SYBASE.bat` file (Windows).
3. Change to `$SYBASE/RAX-15_5/bin` (UNIX or Linux) or `%SYBASE%\RAX-15_5\bin` (Windows).
4. Obtain the Replication Agent version string:
 - For Windows:


```
ra.bat -v
```
 - For UNIX or Linux:


```
./ra.sh -v
```

Note: Make sure permission for the `ra.sh` file is set no broader than 700 (read/write/execute for user, no permissions for group and other).

If Replication Agent installs successfully, you see the SAP copyright and the Replication Agent version string.

Configuring Replication Components

Review the procedures for configuring each of the replication components in the replication system.

See also

- *Testing Replication* on page 110

Setting DB2 UDB Environment Variables

Set the DB2 UDB environment variables.

For UNIX and Linux, the DB2 UDB installation provides two scripts for setting up the DB2 UDB environment variables: `db2cshrc` for C shell and `db2profile` for Bourne or Korn shell. These scripts set the library path environment variable based on the bit size of the installed server or client.

Note: On Windows, the installation sets all necessary environment variables. If Replication Agent is installed on Solaris, AIX or HP Itanium, you must configure a 64-bit DB2 UDB server or client instance.

For UNIX and Linux platforms, the 32-bit and 64-bit versions of the driver and API libraries are located in `$HOME/sqlllib/lib32` and `$HOME/sqlllib/lib64`, respectively, where `$HOME` is the home directory of the DB2 UDB instance owner. If Replication Agent is installed on AIX, Solaris, or HP Itanium, the library path environment variable must point to the 64-bit libraries. For all other platforms, the library path environment variable must point to the 32-bit libraries.

To correctly set the DB2 UDB environment variables for Replication Agent and ECDA, you may need to customize either `db2cshrc` or `db2profile`. However, make any changes to saved copies of the original scripts because the scripts may be changed when you apply a FixPak to the DB2 UDB server or client.

Note: Replication Agent requires 32-bit libraries for Windows and 64-bit libraries for Linux and UNIX. Because ECDA may have different requirements, you may want to maintain two separate customized copies of the aforementioned scripts, one copy for Replication Agent and one for ECDA.

1. Navigate to the DB2 UDB server or client `sqlllib` directory, which is located in the database instance owner user home directory. For example:

```
cd /db2home/db2inst1/sqllib
```

2. Based on the type of UNIX shell you are using, copy either the `db2cshrc` or `db2profile` script:
 - For C shell:


```
cp db2cshrc mydb2cshrc
```
 - For Bourne or Korn shell:


```
cp db2profile mydb2profile
```
3. Edit your copy of the file. Add a statement at the very end of the file that sets the platform-specific library path variable to point to the 32-bit DB2 UDB libraries. For example, on Solaris or Linux, add these lines:
 - For C shell, in `mydb2cshrc`:


```
# force 32-bit libraries
setenv LD_LIBRARY_PATH $HOME/sqllib/lib32:$LD_LIBRARY_PATH
```

where `$HOME` is the home directory of the DB2 UDB instance owner.
 - For Bourne or Korn shell, in `mydb2profile`:


```
# force 32-bit libraries
LD_LIBRARY_PATH=$HOME/sqllib/lib32:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH
```
4. Before starting either Replication Agent, always source your file, if you have customized it. Otherwise, source the original `db2cshrc` or `db2profile` script.

Configuring ECDA Option for ODBC

Review the procedures for configuring ECDA Option for ODBC.

Creating a Maintenance User for Replication in DB2 UDB

The maintenance user is a valid DB2 UDB user that the Replication Server uses to apply commands to the replicate DB2 UDB database.

A user must already exist on the operating system before you can add it to a DB2 UDB database. To add an existing operating system user to the database, use the SAP® Control Center administration tool, which is included with either the server or client installation of DB2 UDB on Microsoft Windows. You may choose to install a DB2 UDB client on a Windows machine and use it to administer the DB2 UDB databases.

1. Select an existing operating system user to be added to the replicate database as a maintenance user.
2. Select **Start > Programs > IBM DB2 > Control Center**.
3. Configure the Control Center administration tool to connect to the replicate DB2 UDB database according to the IBM documentation. Once the Control Center administration tool has been configured, the replicate database appears in the All Databases view.

4. In the Control Center administration tool display window, right-click the replicate database name, and select **Authorities** from the context menu.
5. In the Database Authorities window, click **Add User**.
6. In the Add User window, select the user to add to the database from the list of displayed operating system users, and click **OK**.
7. Click **Grant All** to add necessary authorities to the maintenance user.

Note: In a proof-of-concept environment, the user is created by a DB2 UDB system administrator.

Configuring Replication Server

Review the procedures for configuring Replication Server for DB2 UDB.

Configuring Replication Server for Replication to the Target Data Server

Create a Replication Server connection to the replicate data server.

1. Navigate to the `$SYBASE/RAX-15_5/scripts/ibmudb` (UNIX or Linux) or `%SYBASE%\REP-15_5\scripts\ibmudb` directory.
2. Make a copy of the `udb_create_rs_standby_connection.sql` script named `my_udb_create_rs_standby_connection.sql`.
3. Before executing the `my_udb_create_rs_standby_connection.sql` script against your Replication Server, change the name of the connection that Replication Server uses to connect to the DirectConnect service, where:
 - `{rds}.{rdb}` is the DirectConnect service name followed by the replicate database name.
 - `rs_maint_user` and `rs_maint_user_pwd` are the maintenance user and password created while configuring ECDA Option for ODBC.

For example:

```
create connection to DCService.my_udb_db
using profile rs_udb_to_udb;standard
set username maintuser
set password "password"
go
```

Note: In the previous example, **password** is a Replication Server reserved word and therefore, must be enclosed in double quotes and should match the password specified in the resource file.

4. Create the connection to the replicate database:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i
my_udb_create_rs_standby_connection.sql
```
5. Log in to `SAMPLE_RS`, and verify the Replication Server connection to the standby database:

```
isql -Usa -Psa_pass -SSAMPLE_RS
admin who
go
```

You should see a message returned for the DSI connection *{rds}.{rdb}*. Verify that the status is “Awaiting Message” or “Awaiting Command.”

6. Exit the **isql** session with the **quit** command.

Configuring Replication Server for Replication from the Primary Database

Configure Replication Server for replication from the primary database by granting create object permission to the **rs_username** user.

The **rs_username** user, which is used for communication between Replication Agent and Replication Server, must have **create object** permission before you can use it to create a replication definition. You must grant this permission manually from Replication Server.

1. Log in to Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
```

If you have not set up the sample Replication Server instance, enter your Replication Server instance name in place of **SAMPLE_RS**.

2. Enter:

```
grant create object to SAMPLE_RS_ra
go
```

where *SAMPLE_RS_ra* is the value for the **rs_username** parameter you entered in the resource file.

Creating a Maintenance User on the Primary Connection

The maintenance user on the primary connection is a valid DB2 UDB user whose activities are ignored if Replication Agent processes log records by the user. Replication Server requires one maintenance user to be defined for each connection.

A user must already exist on the operating system before you can add it to a DB2 UDB database. To add an existing operating system user to the database, use the Control Center administration tool, which is included with either the server or client installation of DB2 UDB on Microsoft Windows. You may choose to install a DB2 UDB client on a Windows machine and use it to administer the DB2 UDB databases.

1. Select an existing operating system user to be added to the primary database as a maintenance user.
2. Select **Start > Programs > IBM DB2 > Control Center**.
3. Configure the Control Center administration tool to connect to the primary DB2 UDB database according to the IBM documentation. Once the Control Center administration tool has been configured, the replicate database appears in the All Databases view.

4. In the Control Center administration tool display window, right-click the primary database name, and select **Authorities** from the context menu.
5. In the Database Authorities window, click **Add User**.
6. In the Add User window, select the user to add to the database from the list of displayed operating system users, and click **OK**.
7. Click **Grant All** to add necessary authorities to the maintenance user.

Creating a Replication Server Connection to the Primary Database

Create a Replication Server connection to the primary database.

1. Go to the `$SYBASE/RAX-15_5/scripts/sybase` (UNIX or Linux) or `%SYBASE%\RAX-15_5\scripts\sybase` (Windows) directory.
2. Make a copy of the `rs_create_test_primary_connection.sql` script named `my_udb_rs_create_test_primary_connection.sql`.
3. Before executing the `my_udb_rs_create_test_primary_connection.sql` script against your Replication Server, change all occurrences of `{rax}`.`{test}` to the name of the Replication Server connection between it and Replication Agent for UDB, where:
 - `rax` is the name of the data server of the primary DB2 UDB database.
 - `test` is the name of the database of the primary DB2 UDB.

For example, `NY.NYudb9`.

Note: Save these values to be used later for the `rs_source_ds` and `rs_source_db` parameters in the Replication Agent `ibmudb.rs` file.

4. Change `maint_user` and `maint_pwd` to the user ID and password of the DB2 UDB maintenance user who must have permission to apply DML operations against all user tables to be replicated, where:
 - `maint_user` is the user ID of the DB2 UDB maintenance user you created for the primary connection.
 - `maint_pwd` is the password of the DB2 UDB maintenance user (`maint_user`).

Note: This user must be a valid user in the DB2 UDB database, and it must be different from the user defined by the Replication Agent `pds_username` parameter, which is used for Replication Agent connections to the primary database.

This command creates a Replication Server connection to the primary database example:

```
create connection to NY.NYudb9
set error class rs_udb_error_class
set function string class rs_udb9_function_class
set username maint_user
set password change_on_install
with log transfer on, dsi_suspended
go
```


5. Execute this script in Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i
my_udb_rs_create_test_primary_connection.sql
```

You see a message indicating that the Replication Server connection to the primary database has been created.

See also

- *Creating a Database Replication Definition* on page 99
- *Creating a Database Replication Subscription* on page 100
- *Verifying the ECDA Installed Objects* on page 101

Granting create object Permission to the rs_username User

Grant create object permission to the user ID specified by the Replication Agent **rs_username** parameter.

The **rs_username** user, which is used for communication between Replication Agent and Replication Server, must have **create object** permission before it can be used to create a replication definition. You must grant this permission manually from Replication Server.

1. Log in to Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
```

2. Enter:

```
grant create object to SAMPLE_RS_ra
go
```

where **SAMPLE_RS_ra** is the value for the **rs_username** parameter you entered in the resource file.

3. Connect to the RSSD:

```
1>connect
2>go
```

4. Verify that the **rs_username** user has **create object** permission:

```
1>rs_helpuser
2>go
```

If **rs_username** has **create object** permission, the permission is displayed. For example:

```
SAMPLE_RS_ra connect source, create object
```

Creating a Database Replication Definition

Create a database replication definition.

1. Navigate to the %SYBASE%\RAX-15_5\scripts\sybase (Windows) or \$SYBASE/RAX-15_5/scripts/sybase (UNIX or Linux) directory.

2. Make a copy of the `rs_create_test_db_repdef.sql` script named `my_udb_rs_create_test_db_repdef.sql`.
3. Before executing the `my_udb_rs_create_test_db_repdef.sql` script, change the value of `{pds}`.`{pdb}` to the name of the connection string you defined for the primary database, where:
 - `pds` is the name of the data server of the primary DB2 UDB.
 - `pdb` is the name of database of the primary DB2 UDB.

For example, NY.NYudb9.

Note: Save these values to be used later for the `rs_source_ds` and `rs_source_db` parameters in the Replication Agent `oracle.rs` file. `ibmudb.rs` file.

Here is a database replication definition example:

```
create database replication definition NY_repdef1
with primary at NY.NYudb9
go
```

4. Connect to the Replication Server and enter:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i
my_udb_rs_create_test_db_repdef.sql
```

You see a message indicating that the database replication definition has been created.

See also

- *Creating a Database Replication Subscription* on page 100
- *Verifying the ECDA Installed Objects* on page 101

Creating a Database Replication Subscription

Create a database replication subscription.

1. Make a copy of the `rs_create_test_db_sub.sql` script named `my_udb_rs_create_test_db_sub.sql`.
2. Before executing the `my_udb_rs_create_test_db_sub.sql` script, change the `{pds}`.`{pdb}` and `{rds}`.`{rdb}` to the appropriate connection name, where:
 - `{pds}` – name of the data server of the primary DB2 UDB database.
 - `{pdb}` – name of the database of the primary DB2 UDB database.
 - `{rds}` – name of the DirectConnect service.
 - `{rdb}` – any valid identifier. SAP recommends that you use the DB2 UDB replicate database name.

Note: Save the values of `pds` and `pdb` to be used later for the `rs_source_ds` and `rs_source_db` parameters in the Replication Agent `ibmudb.rs` file.

Here is an example of how to create a database replication subscription:

```
create subscription NY_sub1
for database replication definition NY_repdef1
```

```
with primary at NY.NYudb9
with replicate at DCService.udb_db
without materialization
go
```

3. Connect to the Replication Server and enter:

```
isql -Usa -Psa_pass -SSAMPLE_RS -i
my_udb_rs_create_test_db_sub.sql
```

You see a message indicating that the subscription is being created.

See also

- *Verifying the ECDA Installed Objects* on page 101

Verifying the ECDA Installed Objects

Verify the ECDA installed objects.

1. Log in to DB2 UDB through ECDA as maintuser.

- To verify whether the *rs_info* table has been created successfully, enter:

```
isql -Umaintuser -Ppassword -SDCService
select * from rs_info
go
```

This result is returned:

RSKEY	RSVAL
-----	-----
charset_name	iso_1
sortorder_name	bin_iso_1
(2 rows affected)	

- To verify whether the *rs_lastcommit* table has been created successfully, enter:

```
isql -Umaintuser -Ppassword -SDCService
select * from rs_lastcommit
go
```

This result is returned:

ORIGIN	ORIGIN_QID	SECONDARY_QID	ORIGIN_TIME	DEST_COMMIT_TIME
-----	-----	-----	-----	-----
(0 rows affected)				

If the ECDA objects are not installed successfully, no results appear, and you see an error message.

2. Exit the **isql** session with the **quit** command

Configuring Replication Agent

Review the procedures for configuring Replication Agent for DB2 UDB.

Primary DB2 UDB Configuration for Replication

To configure the primary DB2 UDB database for replication, you may need to connect to the primary database using the primary database instance owner user.

Verifying the Current Archive Setting of the Transaction Log

Set the primary DB2 UDB database transaction logging to archive logging, instead of circular logging for Replication Agent.

1. Determine the LOGARCHMETH1 setting.

- a. Connect to the database by entering this command at the CLP prompt.

```
CONNECT TO dbalias USER db2_user USING db2_user_ps
```

where *dbalias* is the cataloged alias of the primary database, *db2_user* is the primary database user, and *db2_user_ps* is the password.

- b. Determine the LOGARCHMETH1 setting:

```
GET DB CFG FOR dbalias
```

2. If the results do not show that LOGARCHMETH1 is set to LOGRETAIN or to the path name of the directory to which logs are archived, set it:

- To use the default archive location:

```
UPDATE DATABASE CONFIGURATION USING logarchmeth1 LOGRETAIN
```

- To use a specific archive location:

```
UPDATE DATABASE CONFIGURATION USING logarchmeth1 DISK:path
```

where *path* is the full path name of the directory where the archive logs are to be stored.

If you change the setting of the DB2 UDB **logarchmeth1** parameter, DB2 UDB requires you to back up the database. Use your normal backup procedure, or see the IBM documentation for information on the **BACKUP DATABASE** command.

Setting the DB2 UDB Connectivity Autocommit Parameter

Replication Agent requires that the DB2 UDB connectivity autocommit parameter be turned on (autocommit=1).

The autocommit parameter is specified in the DB2 UDB call-level interface (CLI) configuration file for the primary database. If the autocommit parameter is not turned on, a deadlock problem may occur.

To set the autocommit parameter, edit the `db2cli.ini` file. It is located in `<HOME>\sql1lib`, where `<HOME>` is the home directory of the DB2 UDB instance owner (UNIX or Linux) or the DB2 UDB client or server installation directory (Windows).

Creating a DB2 UDB User and Granting Permissions

Create a DB2 UDB user and grant the necessary permissions.

Replication Agent requires a DB2 UDB login that has permission to access data and create new objects in the primary database. The DB2 UDB login must have SYSADM or DBADM authority to access the primary database transaction log.

Note: You can also use these steps to create a maintenance user.

1. Create a new operating system user named *ra_user* using commands appropriate for your operating system.

For example, to create a user named *ra_user* on the UNIX and Linux operating systems, use:

```
useradd -gusers -Gmgmt -s/bin/shell -psybase
-d/home/ra_user -m ra_user
```

where *sybase* is the password corresponding to the *ra_user* user name.

2. Start the DB2 UDB command-line processor:

```
%>bash
%>source /db2home/db2inst1/sqllib/db2profile
%>db2
```

3. Connect to the primary DB2 UDB database:

```
db2=>connect to pdb user db2_admin_user using db2_admin_password
```

where *db2_admin_user* and *db2_admin_password* are the administrative user ID and password for the primary database.

4. Grant all necessary authorities to *ra_user*:

```
db2=>grant DBADM on database to user ra_user
```

Adding a Temporary Tablespace to the Primary Database

Add a temporary tablespace to the primary database.

1. Start the DB2 UDB command-line processor:

```
%>bash
%>source /db2home/db2inst1/sqllib/db2profile
%>db2
```

2. Connect to the primary DB2 UDB database:

```
db2=>connect to pdb user db2_admin_user using db2_admin_password
```

where *db2_admin_user* and *db2_admin_password* are the administrative user ID and password for the primary database.

3. Create a buffer pool:

```
db2=>create bufferpool pdb_buffer_name size 1000 pagesize
same_pagesize_as_pdb
```

where *pdb_buffer_name* is the buffer name, and *same_pagesize_as_pdb* is the page size at the primary database.

4. Create a temporary tablespace:

```
db2=>create user temporary tablespace pdb_temp_space_name
pagesize same_pagesize_as_pdb managed by automatic storage
bufferpool
pdb_buffer_name
```

where *pdb_temp_space_name* is the tablespace name, *same_pagesize_as_pdb* is the page size at the primary database, and *pdb_buffer_name* is the buffer name.

Note: Determine the DB2 UDB page size using the **LIST TABLESPACES SHOW DETAIL** command.

For example, to create a temporary tablespace named *deep13* with a 16KB page size and buffer pool named *tom_servo*, enter:

```
create user temporary tablespace deep13 pagesize 16K managed
by automatic storage bufferpool tom_servo
```

Creating the Replication Agent Instance

Prepare and create a Replication Agent instance for replication of each DB2 UDB database.

A single installation of the Replication Agent can support replication from multiple databases. However, one Replication Agent instance is needed for each DB2 UDB database that is to be replicated.

Note: To replicate from multiple databases, using one Replication Agent instance for each database, you must create a separate DB2 UDB user for each primary database, as described in "Creating a DB2 UDB User and Granting Permissions." You cannot use the same user for multiple Replication Agents.

1. Locate the resource file template.

The majority of configuration values required to create and initialize a Replication Agent can be recorded and stored in a resource file. Using a resource file provides a way to record or retain the configuration information for a Replication Agent instance, allowing an instance to be removed and re-created.

The resource file template for a DB2 UDB instance is `$$SYBASE/RAX-15_5/init/ibmudb.rs` (UNIX or Linux) or `%SYBASE%\RAX-15_5\init\ibmudb.rs` (Windows).

2. Create an instance resource file:

- Copy the resource file template to another file that contains the configuration information for a new instance, for example, *myra.rs*.
- Change the values assigned to properties in the resource file so they match the values required for your environment. The majority of these values are host, port, user, and password information for the DB2 UDB, Replication Server, and RSSD connections.

- Set the value of **rs_source_ds** to the value you chose for pds in previous procedures.
- Set the value of **rs_source_db** to the value you chose for pdb in previous procedures.
- When you create a Replication Agent instance with a resource file:
 - The **asa_password** configuration parameter value must not contain single quotes, double quotes, or a semicolon.
 - The **pds_username** and **pds_password** configuration parameter values must not contain single or double quotes if the **create_pds_username** parameter is set to yes.

The resource file is self-documenting, with each parameter described. See the *Replication Agent Administration Guide*.

Note: This table lists parameters in the order they appear in the resource file, not alphabetically.

Table 6. Resource File Parameters

Parameter	Description	Example values
instance name	Any valid name.	myra
admin_port	Port number that Replication Agent uses.	9030 (if in use, select a different port number)
pds_host_name	Machine (host) where DB2 UDB is installed.	NY
pds_port_number	Port number for DB2 UDB.	50007
pds_database_name	Name of the primary database.	NYudb9
pds_username	User ID that Replication Agent uses to access primary data server.	ra_user
pds_password	Password for pds_username .	sybase
pds_datasource_name	Data source name or database alias for the primary database	DBALIAS
rs_host_name	Machine where Replication Server is installed.	jdoe_host1
rs_port_number	Port where Replication Server is installed.	11752
rs_username	Replication Server user with connect source and create object permissions.	SAMPLE_RS_ra
rs_password	Password for rs_username .	SAMPLE_RS_ra_ps
rs_source_ds	Valid name representing data server of the primary DB2 UDB database.	NY

Parameter	Description	Example values
rs_source_db	Valid name representing the primary DB2 UDB database.	NYudb9
rs_charset	Character set that Replication Server is using. Note: The value defined for the rs_charset configuration parameter must match the RS_charset value in the Replication Server configuration file, \$SYBASE/REP-15_5 /install/<server>.cfg (UNIX or Linux) or %SYBASE%\REP-15_5 \install\<server>.cfg (Windows).	<ul style="list-style-type: none"> Windows: cp850 UNIX: iso_1
rssd_host_name	Machine where RSSD resides.	jdoe_host1
rssd_port_number	Port number where RSSD resides.	11751
rssd_database_name	Database name for RSSD.	SAMPLE_RS_ERSSD
rssd_username	Valid user for RSSD.	SAMPLE_RS_RSSD_maint
rssd_password	Password for rssd_username .	SAMPLE_RS_RSSD_maint_ps
pdb_archive_remove	Enables or disables the removal of archived DB2 UDB transaction log files from the path specified by pdb_archive_path .	USE_DEFAULT
pdb_archive_path	Identifies directory path where Replication Agent expects to find archived DB2 UDB transaction log files.	A valid directory path on the machine hosting Replication Agent that points to a location where DB2 UDB puts the archived transaction log files
start_instance	Start the instance that was created.	yes
initialize_instance	Initialize the Replication Agent instance.	yes

Warning! The **rs_source_ds** and the **rs_source_db** values must match the “pds.pdb” values of your Replication Server primary connection name that you configured in step 3 in the procedure, “Creating a Replication Server connection to the primary database.”

3. Set the DB2 UDB environment variables by sourcing `db2cshrc`, `db2profile`, or your customized versions of the scripts as described in “Setting DB2 UDB environment variables.”
4. Create and execute the new instance resource file:
 - a) Validate the settings in the resource file using the `-vr` parameter.

- On Windows:

```
%SYBASE%\RAX-15_5\bin\ra_admin.bat -vr myra.rs
```

- On UNIX or Linux:

```
$$SYBASE/RAX-15_5/bin/ra_admin.sh -vr myra.rs
```

where *myra.rs* is the path and name of the resource file.

Note: Make sure permission for the *ra_admin.sh* file is set no broader than 700 (read/write/execute for user, no permissions for group and other).

Validation results are returned as either:

- Response-file processing completed
- Or
- Response-file processing completed with errors

If any validation fails, the **ra_admin** utility returns an error message and information about the failure. You can repeat the validation process as many times as necessary until it executes without error. No entities are changed or created.

- b) After the resource file has been validated, allow the **ra_admin** utility to create and configure the Replication Agent instance, using the **-r** parameter.

- On Windows:

```
%SYBASE%\RAX-15_5\bin\ra_admin.bat -r myra.rs
```

- On UNIX or Linux:

```
$$SYBASE/RAX-15_5/bin/ra_admin.sh -r myra.rs
```

where *myra.rs* is the path and name of the resource file.

Note: If, in your response file, you set **start_instance** to **yes**, your instance is also running. If you set **initialize_instance** to **yes**, your instance is also initialized.

- c) Creation results are returned as either:

- Response-file processing completed
- Or
- Response-file processing completed with errors

See the *Replication Agent Administration Guide*.

5. If, in your resource file, you set **start_instance** to **yes**, skip this step, and continue with the next section. If your resource file did not configure **start_instance** to **yes**, change to the instance directory and run Replication Agent in the background.

- On Windows:

```
cd %SYBASE%\RAX-15_5\myra
```

- On UNIX or Linux:

```
cd $$SYBASE/RAX-15_5/myra
```

Execute the RUN file in the background.

Your Replication Agent for DB2 UDB is now running.

See also

- *Verifying the Replication Agent Instance Installation* on page 108
- *Initializing the Replication Agent Instance* on page 109

Verifying the Replication Agent Instance Installation

Verify that the Replication Agent instance has been successfully installed.

1. Use **dsedit** to update the Replication Server `sql.ini` (Windows) or `interfaces` (UNIX or Linux) file to include an entry for the Replication Agent location.

Note: You can use any Tabular Data Stream™ (TDS) client utility (**isql**, **isqlApp**, or **SQLAdvantage**) that you prefer.

2. Verify the connection to Replication Agent:

- a) Open a command window in the `%SYBASE%` (Windows) or `$SYBASE` (UNIX or Linux) directory of your Replication Agent installation.
- b) Set the environment variables by executing the `SYBASE.bat` file (Windows) or by sourcing the `SYBASE.csh` file (UNIX or Linux).
- c) Log in to Replication Agent:

```
isql -Usa -Psa_pass -Smyra
```

Note: These verification steps are optional because they were performed when you verified the resource file.

3. Verify the Replication Agent connection to Replication Server:

- a) Enter:

```
test_connection RS
go
```

- b) This result is returned:

```
Type Connection
-----
RS      succeeded
(1 row affected)
```

- c) If the result indicates a failure, either the server is not responding or the connection properties (*host*, *port*, *user*, or *password*) are incorrect. Verify the host and port configuration values, and manually log in to the Replication Server as the configured user to determine which property is incorrectly defined.

See the *Replication Agent Administration Guide*.

4. Verify the Replication Agent connection to the primary database:

- a) Enter:

```
test_connection PDS
go
```

You see:

```
Type Connection
-----
PDS succeeded
(1 row affected)
```

b) If the result indicates a failure:

- The server is not responding, or
- One of the connection properties is incorrect.

Manually log in to the primary DB2 UDB database as the configured user to find which property is incorrectly defined.

See *Testing Network Connectivity* in the *Replication Agent Administration Guide*.

See also

- *Initializing the Replication Agent Instance* on page 109

Initializing the Replication Agent Instance

Initialize the Replication Agent instance.

The **ra_admin init** command verifies that the primary DB2 UDB database is correctly configured to provide archived logging and that the DB2 UDB user ID used by the Replication Agent has the necessary permissions. The **ra_admin init** command also creates objects in the primary database to support replication.

1. Initialize the Replication Agent instance:

```
ra_admin init
go
```

You see a message indicating that the procedure was successful.

2. Enter:

```
resume
go
```

If the Replication Agent successfully transfers to a replicating state, you see:

```
State          Action
-----
REPLICATING   Ready to replicate data.
```

3. The **ra_status** command returns the state of the Replication Agent. It is good practice to verify that the Replication Agent remains in replication state, even after the resume command executes successfully.

To detect an error that occurred after replication start-up, execute:

```
ra_status
go
```

If the Replication Agent is in replicating state, **ra_status** returns:

```
State          Action
-----
REPLICATING   Ready to replicate data.
```

4. Validate that both primary and replicate connections are active:

```
isql -Usa -Psa_pass -SSAMPLE_RS
admin who
go
```

Note: Be aware that:

- The DSI connection for the primary database connection is usually suspended because you are not replicating data back to the primary database.
- The Replication Agent connection, if established for the replicate database connection, is usually down, because you are not replicating data from the replicate database.

Do not proceed until **admin who** returns thread status similar to:

```
Spid Name          State              Info
-----
13  DSI EXEC         Awaiting Command  101(1) SAMPLE_RS_ERSSD.
SAMPLE_RS_ERSSD
9   DSI              Awaiting Message  101 SAMPLE_RS_ERSSD.
SAMPLE_RS_ERSSD
8   SQM              Awaiting Message  101:0 SAMPLE_RS_ERSSD.
SAMPLE_RS_ERSSD
54  DSI EXEC         Awaiting Command  102(1) DCService.rdb
53  DSI              Awaiting Message  102 DCService.rdb
17  SQM              Awaiting Message  102:0 DCService.rdb
   DSI EXEC         Suspended         103(1) pds.pdb
   DSI              Suspended         103 pds.pdb
24  DIST            Awaiting Wakeup   103 pds.pdb
25  SQT              Awaiting Wakeup   103:1 DIST pds.pdb
23  SQM              Awaiting Message  103:1 pds.pdb
22  SQM              Awaiting Message  103:0 pds.pdb
62  REP AGENT       Awaiting Command  pds.pdb
```

Testing Replication

When you finish setting up the replication system, test that replication works as intended.

1. Connect to the primary DB2 UDB instance as a regular user, rather than as the maintenance user. Make sure the regular user also exists in the replicate database.
 - a) To connect to the database with CLP, use:

```
CONNECT TO dbalias USER db2_user USING db2_user_ps
```

where *dbalias* is the cataloged alias of the primary database, *db2_user* is the primary database user, and *db2_user_ps* is the password.

2. Create a test table to replicate in the primary database:

```
create table T1 (a int, b char(10))
```

- Grant permissions to any new or existing object to be replicated in the primary database:

```
grant all on T1 to public
```

- Connect to the replicate DB2 UDB instance as a regular user, rather than as the maintenance user.

- Create a test table in the replicate database:

```
create table T1 (a int, b char(10))
```

- Grant permissions to any new or existing object to be replicated in the replicate database so that the Replication Server maintenance user can update this table:

```
grant all on T1 to public
```

- Connect to the Replication Agent and resume replication:

```
resume
```

- Mark the table:

```
pdb_setreptable T1, mark
```

- After the table is marked and the Replication Agent is in a replicating state, insert test data into the test table and commit it.

By default, the work performed by the maintenance user of the primary connection is not replicated. The user ID used to insert transactions cannot be the same as the maintenance user defined in the primary connection.

- Use your preferred query tool to examine the replicate site for results and compare the contents of your test table from both the primary and replicate sites.

Resetting the Primary DB2 UDB Database for Replication

In a test environment, you may occasionally need to reset the replication environment.

- Log in to the Replication Agent and issue:

```
isql -Usa -Psa_pass -Smyra
suspend
go
```

- Reset the locator stored in the Replication Agent:

```
ra_locator zero
go
```

- Reset the locator stored in Replication Server:

```
isql -Usa -Psa_pass -SSAMPLE_RS
connect
go
rs_zeroltm pds, pdb
go
exit
```

You see:

CHAPTER 5: DB2 UDB Replication Setup

```
Locator has been reset to zero.  
(return status = 0)
```

4. Resume replication with the Replication Agent **resume** command:

```
isql -Usa -Psa_pass -Smyra  
1> resume  
2> go
```

Replication Server, Replication Agent, and database users.

Table 7. Replication Server Users

User/Login	Server	Client	Usage
rs_user	Replication Server	Replication Server administrators and users	Executes commands on the replication server. The commands executed depend on the security level of the user and include commands such as creating replication definitions, subscriptions, and creating other users.
Replication Server login username rs_username	Replication Server	Replication Server users and Replication Agent	Replication Agent uses the rs_user user to send commands to the Replication Server. For example, replication definition commands and LTL commands. This user must have the permissions to send LTL, create, and drop replication definitions.
Primary database maintenance user	Primary database	Replicate Server	Executes DML at the primary database in the case of bi-directional replication. Use the maintenance user if you do not want the operations at the primary database to be sent to the replicate database. This user cannot be the same as the pds_username user.
Replicate database maintenance user	Replicate database	Replication Server	Executes DDL, DML, and stored procedures at the replicate database. The transactions applied by the maintenance user are not replicated out of the replicate database. This user cannot be the same as the pds_username user.

User/Login	Server	Client	Usage
RSSD primary user rssd_username	Replication Server	Replication Agent	<p>Replication Agent uses this user to login to the Replication Server RSSD for replication definition information.</p> <p>Replication Server provides a gateway for access to the RSSD. Replication Server users other than the Replication Agent can access the RSSD using the rs_username login. This user cannot be the same as the pds_username user.</p>
RSSD maintenance user	Replicate RSSD	Replicate, Replication Server	Applies modifications to replicate RSSDs to coordinate with route information in the primary RSSD.

Table 8. Replication Agent Users

User/Login	Server	Client	Usage
pds_username	Primary database	Replication Agent	<p>Replication Agent uses this user login name to access primary database objects and to create, remove, or manage the transaction log objects in the primary database. The login name cannot be the same as the Replication Agent user for RSSD or the maintenance user of the primary database.</p> <p>For Replication Agents for Oracle, IBM DB2 UDB, and Microsoft SQL Server, the login name selects the data if you use direct load materialization.</p> <p>For Replication Agent for IBM DB2 UDB, this user must have sysdba permission to read DB2 logs and these permissions:</p> <ul style="list-style-type: none"> • SELECT permissions for reading the log. • SELECT permissions on the required system and user tables. <p>pds_username cannot be the same as the primary database maintenance user or the RSSD primary user.</p>

User/Login	Server	Client	Usage
Replication Agent administrator ra_username	Replication Agent	Replication Agent user and administrator	Administers the Replication Agent instance.

Table 9. Database Users

User/Login	Server	Client	Usage
Data server login	Primary database	Primary database user	Data server login names are used with a client application to connect to a data server. This is the login used by the application or user whose activities in the primary database (for example, DML, DDL, and stored procedures) are to be replicated.
Primary database administrative user	Primary database	Primary database user and Replication Agent	Creates other users in the primary database and performs configuration activities such as granting permissions to database objects. The Replication Agent for IBM DB2 UDB requires this user to read the DB2 log. The Replication Agent for Microsoft SQL Server required this user to add system stored procedures.
ASM username asm_username	Primary database	Replication Agent	(Oracle only) Replication Agent uses this user to access the Oracle ASM server.
ddl_username	Replicate database	Replication Server	The ddl_username user has the permissions to perform tasks on behalf of other users. This allows consistent object ownership across primary and replicate database.

CHAPTER 7 **Glossary**

This glossary describes SAP® Replication Server® Options terms.

- **SAP Adaptive Server** – the brand name for SAP relational database management system (RDBMS) software products.
 - SAP® Adaptive Server® Enterprise manages multiple, large relational databases for high-volume online transaction processing (OLTP) systems and client applications.
 - SAP® IQ manages multiple, large relational databases with special indexing algorithms to support high-speed, high-volume business intelligence, decision support, and reporting client applications.
 - SAP® SQL Anywhere® manages relational databases with a small DBMS footprint, which is ideal for embedded applications and mobile device applications.

See also *DBMS* and *RDBMS*.

- **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 *bulk materialization* and *nonatomic materialization*.
- **BCP utility** – a feature that improves SAP Replication Server performance when replicating large batches of insert statements on the same table in SAP Adaptive Server Enterprise supported versions. 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 SAP® Open Client™/Open Server™ Bulk-Library. See also *bulk copy*.
- **bulk copy** – an SAP® Open Client™ interface for the high-speed transfer of data between a database table and program variables. Bulk copying provides an alternative to using SQL **insert** and **select** commands to transfer data.
- **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* and *nonatomic materialization*.
- **client** – in client/server systems, the part of the system that sends requests to servers and processes the results of those requests. See also *client application*.
- **client application** – software that is responsible for the user interface, including menus, data entry screens, and report formats. See also *client*.

- **commit** – an instruction to the DBMS to make permanent the changes requested in a transaction. See also *transaction*. Contrast with *rollback*.
- **database** – a collection of data with a specific structure (or schema) for accepting, storing, and providing data for users. See also *data server*, *DBMS*, and *RDBMS*.
- **database connection** – a connection that allows SAP Replication Server or Replication Agent to manage the database and send or receive transactions from or to a database. Each database in a replication system can have only one database connection in SAP Replication Server and one primary database connection in Replication Agent. See also *SAP Replication Server* and *route*.
- **data client** – a client application that provides access to data by connecting to a data server. See also *client*, *client application*, and *data server*.
- **data distribution** – a method of locating (or placing) discrete parts of a single set of data in multiple systems or at multiple sites. Data distribution is distinct from data replication, although a data replication system can be used to implement or support data distribution. Contrast with *data replication*.
- **data replication** – the process of copying primary data to remote locations and synchronizing the copied data with the primary data. Data replication is different from data distribution. Replicated data is a stored copy of data at one or more remote sites throughout a system, and it is not necessarily distributed data. Contrast with *data distribution*. See also *transaction replication*.
- **data server** – a server that provides the functionality necessary to maintain the physical representation of a table in a database. Data servers are usually database servers, but they can also be any data repository with the interface and functionality a data client requires. See also *client*, *client application*, and *data client*.
- **datatype** – a keyword that identifies the characteristics of stored information on a computer. Some common datatypes are: *char*, *int*, *smallint*, *date*, *time*, *numeric*, and *float*. Different data servers support different datatypes.
- **DBMS** – database management system

a computer-based system for defining, creating, manipulating, controlling, managing, and using databases. The DBMS can include the user interface for using the database, or it can be a standalone data server system. Compare with *RDBMS*.
- **Direct Load Materialization** – an automatic materialization method to consume data between different primary and replicate databases. Data is loaded directly from a primary table into a replicate table, no materialization queue is used. See also *materialization*, *atomic materialization*, *bulk materialization*, and *nonatomic materialization*.
- **ERSSD** – Embedded Replication Server System Database

The SAP SQL Anywhere database that stores Replication Server system tables. You can choose whether to store SAP Replication Server system tables on the ERSSD or the SAP Adaptive Server Enterprise RSSD. See also *SAP Replication Server*.

- **failback** – a procedure that restores the normal user and client access to a primary database, after a failover procedure switches access from the primary database to a replicate database. See also *failover*.
- **failover** – a procedure that switches user and client access from a primary database to a replicate database, particularly in the event of a failure that interrupts operations at the primary database, or access to the primary database. Failover is an important fault-tolerance feature for systems that require high availability. See also *failback*.
- **function** – a data server object that represents an operation or set of operations. SAP Replication Server distributes operations to replicate databases as functions. See also *stored procedure*.
- **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. See also *function*.
- **gateway** – connectivity software that allows two or more computer systems with different network architectures to communicate.
- **inbound queue** – a stable queue managed by SAP Replication Server to spool messages received from a Replication Agent. See also *outbound queue* and *stable queue*.
- **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 Adaptive Servers, gateways, Replication Servers, and Replication Agents. The interfaces file entries enable clients and servers to connect to each other in a network. See also *SAP Open Client* and *SAP Open Server*.
- **isql** – an Interactive SQL client application that can connect and communicate with any SAP® Open Server™ application, including SAP Adaptive Server Enterprise, Replication Agent, and SAP Replication Server. See also *SAP Open Client* and *SAP Open Server*.
- **Java** – an object-oriented programming language developed by Oracle (formerly Sun Microsystems). A platform-independent, “write once, run anywhere” programming language.
- **JVM** – Java Virtual Machine

the JVM is the part of the Java Runtime Environment (JRE) that is responsible for interpreting Java byte codes. See also *Java* and *JRE*.
- **JDBC** – Java Database Connectivity

JDBC is the standard communication protocol for connectivity between Java clients and data servers. See also *data server* and *Java*.
- **JRE** – Java Runtime Environment

the JRE consists of the Java Virtual Machine (JVM), the Java Core Classes, and supporting files. The JRE must be installed on a machine to run Java applications, such as Replication Agent. See also *Java VM*.
- **LAN** – local area network

a computer network located on the user premises and covering a limited geographical area (usually a single site). Communication within a local area network is not subject to

external regulations; however, communication across the LAN boundary can be subject to some form of regulation. Contrast with *WAN*.

- **latency** – in transaction replication, the time it takes to replicate a transaction from a primary database to a replicate database. Specifically, latency is the time elapsed between committing an original transaction in the primary database and committing the replicated transaction in the replicate database.

In disk replication, latency is the time elapsed between a disk write operation that changes a block or page on a primary device and the disk write operation that changes the replicated block or page on a replicate device.

See also *transaction replication*.

- **LOB** – large object

a large collection of data stored as a single entity in a database.

- **Log Reader** – an internal component of Replication Agent that interacts with the primary database to capture transactions for replication. See also *Log Transfer Interface* and *LTM*.
- **LTI** – Log Transfer Interface

an internal component of Replication Agent that interacts with SAP Replication Server to forward transactions for distribution to Replication Server. See also *Log Reader* and *LTM*.

- **LTL** – Log Transfer Language

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. See also *Log Reader* and *LTI*.

- **LTM** – Log Transfer Manager

an internal component of Replication Agent that interacts with the other Replication Agent internal components to control and coordinate Replication Agent operations. See also *Log Reader* and *LTI*.

- **maintenance user** – a special user login name in the replicate database that SAP Replication Server uses to apply replicated transactions to the database. See also *replicate database* and *SAP Replication Server*.
- **materialization** – the process of copying the data from a primary database to a replicate database, initializing the replicate database so that the replication system can begin replicating transactions. See also *atomic materialization*, *bulk materialization*, and *nonatomic materialization*.
- **MPR** – 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.

- **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. Contrast with *atomic materialization*. See also *bulk materialization*.
- **ODBC** – Open Database Connectivity
an industry-standard communication protocol for clients connecting to data servers. See also *client*, *data server*, and *JDBC*.
- **SAP Open Client** – an SAP product that provides customer applications, third-party products, and other SAP products with the interfaces needed to communicate with SAP Open Server applications. See also *SAP Open Server*.
- **Open Client application** – An application that uses SAP Open Client libraries to implement Open Client communication protocols. See also *SAP Open Client* and *SAP Open Server*.
- **SAP Open Server** – an SAP product that provides the tools and interfaces required to create a custom server. See also *SAP Open Client*.
- **Open Server application** – a server application that uses SAP Open Server libraries to implement Open Server communication protocols. See also *SAP Open Client* and *SAP Open Server*.
- **outbound queue** – a stable queue managed by SAP Replication Server to spool messages to a replicate database. See also *inbound queue*, *replicate database*, and *stable queue*.
- **primary data** – the data source used for replication. Primary data is stored and managed by the primary database. See also *primary database*.
- **primary database** – the database that contains the data to be replicated to another database (the replicate database) through a replication system. The primary database is the source of replicated data in a replication system. Sometimes called the active database. Contrast with *replicate database*. See also *primary data*.
- **primary key** – a column or set of columns that uniquely identifies each row in a table.
- **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 also *primary database* and *replicate site*.
- **primary table** – a table used as a source for replication. Primary tables are defined in the primary database schema. See also *primary data* and *primary database*.
- **primary transaction** – a transaction that is committed in the primary database and recorded in the primary database transaction log. See also *primary database*, *replicated transaction*, and *transaction log*.

- **quiesce** – to cause a system to go into a state in which further data changes are not allowed. See also *quiescent*.
- **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.

In a database, a state in which all data updates are suspended so that transactions cannot change any data, and the data and log devices are stable.

This term is interchangeable with quiesced and in quiesce. See also *quiesce*.

- **RASD** – Replication Agent System Database

Information in the RASD is used by the primary database to recognize database structure or schema objects in the transaction log.

- **RCL** – Replication Command Language

the command language used to manage SAP Replication Server. See also *SAP Replication Server*.

- **RDBMS** – relational database management system

an application that manages and controls relational databases. Compare with *DBMS*. See also *relational database*.

- **relational database** – a collection of data in which data is viewed as being stored in tables, which consist of columns (data items) and rows (units of information). Relational databases can be accessed by SQL requests. Compare with *database*. See also *SQL*.
- **replicate data** – A set of data that is replicated from a primary database to a replicate database by a replication system. See also *primary database*, *replication system*, and *replicate database*.
- **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*. See also *replicate data*, *replicated transaction*, and *replication system*.
- **replicated transaction** – a primary transaction that is replicated from a primary database to a replicate database by a transaction replication system. See also *primary database*, *primary transaction*, *replicate database*, and *transaction replication*.
- **replicate site** – the location or facility at which replicate data servers and replicate databases are deployed to support normal business operations during scheduled downtime at the primary site. Contrast with *primary site*. See also *replicate database*.
- **Replication Agent** – an application that reads a primary database transaction log to acquire information about data-changing transactions in the primary database, processes the log information, and then sends it to an SAP Replication Server for distribution to a replicate database. See also *primary database* and *SAP Replication Server*.
- **replication definition** – a description of a table or stored procedure in a primary database, 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 table or stored procedure. See also *SAP Replication Server* and *subscription*.

- **replication system** – a data processing system that replicates data from one location to another. Data can be replicated between separate systems at a single site, or from one or more local systems to one or more remote systems. See also *transaction replication*.
- **rollback** – an instruction to a database to back out of the changes requested in a unit of work (called a transaction). Contrast with *commit*. See also *transaction*.
- **route** – A one-way message stream from a primary SAP Replication Server to a replicate SAP Replication Server. Routes carry data-changing commands (including those for RSSDs) and replicated functions (database procedures) between separate SAP Replication Servers. See also *SAP Replication Server*.
- **RSSD** – Replication Server System Database

the SAP Adaptive Server Enterprise (SAP ASE) database containing an SAP Replication Server system tables. The user can choose whether to store Replication Server system tables on SAP ASE or embedded in an SAP SQL Anywhere database hosted by SAP Replication Server. See also *SAP Replication Server*.

- **SAP Replication Server** – an SAP software product that provides the infrastructure for a transaction replication system. See also *Replication Agent*.
- **SQL** – Structured Query Language

a nonprocedural programming language used to process data in a relational database. ANSI SQL is an industry standard. See also *transaction*.

- **stable queue** – a disk device-based, store-and-forward queue managed by SAP Replication Server. Messages written into the stable queue remain there until they can be delivered to the appropriate process or replicate database. SAP Replication Server provides a stable queue for both incoming messages (the inbound queue) and outgoing messages (the outbound queue). See also *database connection*, *SAP Replication Server*, and *route*.
- **stored procedure** – a data server object that represents an operation or set of operations. This term is often used interchangeably with *function*.
- **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. See also *replicate database*, *replication definition*, and *SAP Replication Server*.
- **table** – in a relational DBMS, a two-dimensional array of data or a named data object that contains a specific number of unordered rows composed of a group of columns that are specific for the table. See also *database*.
- **transaction** – a unit of work in a database that can include zero, one, or many operations (including **insert**, **update**, and **delete** operations), and that is either applied or rejected as a whole. Each SQL statement that modifies data can be treated as a separate transaction, if the database is so configured. See also *SQL*.

CHAPTER 7: Glossary

- **transactional consistency** – A condition in which all transactions in the primary database are applied in the replicate database, and in the same order that they were applied in the primary database.
- **transaction log** – generally, the log of transactions that affect the data managed by a data server. Replication Agent reads the transaction log to identify and acquire the transactions to be replicated from the primary database. See also *Replication Agent, primary database*, and *SAP Replication Server*.
- **transaction replication** – a data replication method that copies data-changing operations from a primary database to a replicate database. See also *data replication*.
- **UDB** – IBM DB2 Universal Database (formerly IBM DB2 for Linux, UNIX, and Windows).
- **WAN** – wide-area network
a system of local-area networks (LANs) connected together with data communication lines. Contrast with *LAN*.

Index

A

accounts, creating sybase user 10

B

before you begin 5

C

configuring

- ECDA Option for ODBC 59, 95
- Microsoft SQL Server ODBC driver 59
- Primary Oracle database for replication 30
- Replication Agent for DB2 UDB 101
- Replication Agent for Microsoft SQL Server 67

- Replication Agent for Oracle 30
- replication components 23
- replication components for Microsoft SQL Server 58
- Replication Server for Microsoft SQL Server 62
- Replication Server for Oracle 24
- Replication Server for replication to standby 62

configuring ExpressConnect for Oracle 28

configuring Replication Agent

- configuring and validating the resource file 71

- configuring the primary DB2 UDB database for replication 102

- creating a Microsoft SQL Server user and granting permissions 67

- disabling Microsoft SQL Server replication 67

- enabling remote DAC 67

- first-time initialization 74

- obtaining Microsoft SQL Server DAC port number 68

- setting the Autocommit parameter 102

- subsequent initialization 76

- verifying compatibility level of Microsoft SQL Server 67

- verifying current archive setting of transaction log 102

- verifying the Replication Agent replicating state 77

configuring Replication Server

- for replication from the primary database 24, 62

- for replication to standby 24

conventions

- style 1
- syntax 1

creating

- database replication definition 65
- database replication definition for DB2 UDB 99

- database replication definitions 27

- database replication subscription for DB2 UDB 100

- database replication subscriptions 28, 66

- DirectConnect instance 60

- maintenance user 61, 95

- Oracle user 33

- Replication Agent instance 36

- Replication Agent instance in DB2 UDB 104

- Replication Server connection to standby database 24, 98

- Replication Server connection to standby database for Microsoft SQL Server 62

- Replication Server from primary database 25, 64

creating sybase user accounts 10

D

DB2 UDB

- adding a temporary tablespace to the primary database 103

- configuring Replication Server 96

- configuring Replication Server for replication from the primary database 97

- configuring Replication Server for replication to replicate database 96

- creating a database replication definition 99

- creating a database replication subscription 100

- creating a Replication Server connection to standby database 98

Index

- creating a user and granting permissions 103
 - creating the Replication Agent instance 104
 - initializing the Replication Agent instance 109
 - installing replication components 84
 - replication components 81
 - replication overview 81
 - resetting the primary database for replication 111
 - setting environment variables 94
 - testing DB2 UDB replication 110
 - testing replication 110
 - verifying the ECDA installed objects 101
 - verifying the Replication Agent instance 108
- DB2 UDB prerequisites 82
- dsedit utility
- adding server entries to interfaces file 9
- ### E
- errors
- while starting installer 16, 54, 88
- ExpressConnect for Oracle
- configuring 28
- ### I
- initializing
- Replication Agent instance 42
 - Replication Agent instance in DB2 UDB 109
- installation guidelines
- for ECDA Option for ODBC 50, 84
 - for Replication Agent 21, 57, 91
 - for Replication Server 15, 53, 87
- installation program
- Replication Agent 21, 57, 91
- installer
- errors in starting 16, 54, 88
- installing 71
- installing ECDA Option for ODBC 50, 84
 - installing Oracle Instant Client libraries 19
 - installing Replication Server 15, 53, 87
- interfaces file 9
- ### L
- LogMiner
- configuration 31
- ### M
- Microsoft SQL Server
- configuring and verifying the primary database 67
 - configuring Replication Server 62
 - configuring Replication Server for replication from the primary database 63
 - creating a DirectConnect instance for 60
 - creating a maintenance user for 61
 - resetting the primary database for replication 79
 - system requirements 47
 - testing replication 49, 78
 - verifying the Replication Agent instance 75
- Microsoft SQL Server JDBC driver 71
- ### O
- Oracle
- configuring Replication Server 24
 - configuring Replication Server for replication from the primary database 25
 - installing replication components 15
 - system requirements 13
 - testing replication 44
 - verifying the Replication Agent instance 41
- Oracle Instant Client libraries
- installing 19
- ### P
- planning your installation
- replication components 12, 48, 82
- prerequisites 48, 83
- Oracle 13
- prerequisites for installing replication components 48
- ### R
- replication
- components 11, 47
 - overview 11, 47
- Replication Agent
- installing 21, 57, 91
- Replication Agent for DB2 UDB
- configuring 101
- Replication Agent for Microsoft SQL Server
- configuring 67
- Replication Agent for Oracle
- configuring 30
- Replication Agent installation directory 14, 49, 83

- replication components
 - configuration 58, 94
- replication from the primary Microsoft SQL Server database
 - configuring Replication Server 62
- Replication Server
 - verifying installation 19, 91
- replication system
 - DB2 UDB diagram 81
 - diagram of Microsoft SQL Server 47
- resetting primary Oracle database for replication 45

S

- setting up replication for DB2 UDB 81
- setting up replication for Oracle 11
- setup prerequisites 12, 47, 81
- sybase user account
 - creating 10
- system requirements 12
 - Microsoft SQL Server 47

T

- testing
 - Microsoft SQL Server replication 49, 78

- Oracle replication 44

U

- user account
 - sybase 10

V

- verify connection to the replicate Microsoft SQL Server 61
- verifying
 - current archive setting of the redo logs 30
 - installation 23, 56, 58, 93
 - primary Oracle configuration for replication 35
 - Replication Agent instance in DB2 UDB 108
 - Replication Agent instance in Microsoft SQL Server 75
 - Replication Agent instance in Oracle 41
 - supplemental logging of primary key data 30

