SYBASE®

Quick Start Guide

Replication Server® Options

15.1

[Linux, Microsoft Windows, and UNIX]

DOCUMENT ID: DC00766-01-1510-01

LAST REVISED: February 2008

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

Contents

About This Book

Replication Server Options extend the capabilities of Replication Server® to support replication to and from the following primary data servers in a Sybase replication system:

- IBM DB2 Universal Database (for UNIX and Microsoft Windows)
- Microsoft SQL Server
- Oracle Database Server

The Replication Server Options consist of the following components:

- Replication Agent™
- Enterprise Connect Data Access™ (ECDA)

Audience

This book is for anyone who needs to manage or administer a Sybase replication system with non-Sybase primary databases, or administer the non-Sybase primary databases in a Sybase replication system. This may include:

- Database Administrators
- Network Administrators
- System Administrators

How to use this book

Use the product documentation to look up detailed information about Replication Server, Replication Agent, and ECDA commands and configuration parameters.

This book is organized as follows:

Chapter 1, “Setting Up Replication for Microsoft SQL Server,” describes the quick-start procedure to install and configure a sample replication environment for Microsoft SQL Server.

Chapter 2, “Setting Up Replication for Oracle,” describes the quick-start procedure to install and configure a sample replication environment for Oracle.

Chapter 3, “Setting Up Replication for IBM DB2 Universal Database,” describes the quick-start procedure to install and configure a sample replication environment for IBM DB2 Universal Database.

Related documents

Replication Server, Replication Agent, and ECDA See the product documentation for details on how to install and use Replication Server and the Replication Server Options components.

Primary data server Make sure that you have appropriate documentation for the primary data server that you use with the Sybase replication system.

Java environment The Replication Agent requires a Java Runtime Environment (JRE) on the Replication Agent host machine.

- The Replication Agent *Release Bulletin* contains the most up-to-date information about Java and JRE requirements.
- Java documentation available from your operating system vendor describes how to set up and manage the Java environment on your platform.

Other sources of information

Use the Sybase Getting Started CD, the SyBooks™ CD, and the Sybase Product Manuals Web site to learn more about your product:

- The Getting Started CD contains release bulletins and installation guides in PDF format, and may also contain other documents or updated information not included on the SyBooks CD. It is included with your software. To read or print documents on the Getting Started CD, you need Adobe Acrobat Reader, which you can download at no charge from the Adobe Web site using a link provided on the CD.
- The SyBooks CD contains product manuals and is included with your software. The Eclipse-based SyBooks browser allows you to access the manuals in an easy-to-use, HTML-based format.

  Some documentation may be provided in PDF format, which you can access through the PDF directory on the SyBooks CD. To read or print the PDF files, you need Adobe Acrobat Reader.

  Refer to the *SyBooks Installation Guide* on the Getting Started CD, or the *README.txt* file on the SyBooks CD for instructions on installing and starting SyBooks.
- The Sybase Product Manuals Web site is an online version of the SyBooks CD that you can access using a standard Web browser. In addition to product manuals, you will find links to EBFs/Maintenance, Technical Documents, Case Management, Solved Cases, newsgroups, and the Sybase Developer Network.

To access the Sybase Product Manuals Web site, go to Product Manuals at http://www.sybase.com/support/manuals/.

Sybase certifications on the Web

Technical documentation at the Sybase Web site is updated frequently.

❖ Finding the latest information on product certifications

1. Point your Web browser to Technical Documents at http://www.sybase.com/support/techdocs/.
2. Click Certification Report.
3. In the Certification Report filter select a product, platform, and timeframe and then click Go.
4. Click a Certification Report title to display the report.

❖ Finding the latest information on component certifications

1. Point your Web browser to Availability and Certification Reports at http://certification.sybase.com/.
2. Either select the product family and product under Search by Base Product; or select the platform and product under Search by Platform.
3. Select Search to display the availability and certification report for the selection.

❖ Creating a personalized view of the Sybase Web site (including support pages)

Set up a MySybase profile. MySybase is a free service that allows you to create a personalized view of Sybase Web pages.

1. Point your Web browser to Technical Documents at http://www.sybase.com/support/techdocs/.
2. Click MySybase and create a MySybase profile.

Sybase EBFs and software maintenance

❖ Finding the latest information on EBFs and software maintenance

1. Point your Web browser to the Sybase Support Page at http://www.sybase.com/support.
2. Select EBFs/Maintenance. If prompted, enter your MySybase user name and password.
3. Select a product.

4. Specify a time frame and click Go. A list of EBF/Maintenance releases is displayed.

   Padlock icons indicate that you do not have download authorization for certain EBF/Maintenance releases because you are not registered as a Technical Support Contact. If you have not registered, but have valid information provided by your Sybase representative or through your support contract, click Edit Roles to add the “Technical Support Contact” role to your MySybase profile.

5. Click the Info icon to display the EBF/Maintenance report, or click the product description to download the software.

Style conventions

The following style conventions are used in this book:

- In a sample screen display, commands that you should enter exactly as shown appear like this:

  ```
  ra_init
  ```

- In the regular text of this document, variables or user-supplied words appear like this:

  Specify the *value* option to change the setting of the configuration parameter.

- In a sample screen display, variables or words that you should replace with the appropriate value for your site appear like this:

  ```
  resume connection to pds.pdb
  ```

  Here, *pds* and *pdb* are the variables you should replace.

- In the regular text of this document, names of programs, utilities, procedures, and commands appear like this:

  Use the pdb_init command to initialize the primary database.

- In the regular text of this document, names of database objects (tables, columns, stored procedures, etc.) appear like this:

  Check the price column in the widgets table.

- In the regular text of this document, names of datatypes appear like this:

  Use the date or datetime datatype.

- In the regular text of this document, names of files and directories appear like this:

  Log files are located in the *$SYBASE/RAX-15_1/inst_name/log* directory.

Syntax conventions

The following syntax conventions are used in this book:

Table 1: 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.

In reference sections of this document, statements that show the syntax of commands appear like this:

ra_config [*param* [, *value*]]

The words *param* and *value* in the syntax are variables or user-supplied words.

Character case conventions

The following character case conventions are used in this book:

- All command syntax and command examples are shown in lowercase. However, Replication Agent 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 Agent commands. However, if you need to use a mixed-case object name in a Replication Agent command (to match a mixed-case object name in the primary database), you must delimit the object name with quote characters. For example:

```
pdb_get_tables "TableName"
```

Accessibility features

This document is available in an HTML version that is specialized for accessibility. You can navigate the HTML with an adaptive technology such as a screen reader, or view it with a screen enlarger.

Replication Server Options 15.1 and the HTML documentation have been tested for compliance with U.S. government Section 508 Accessibility requirements. Documents that comply with Section 508 generally also meet non-U.S. accessibility guidelines, such as the World Wide Web Consortium (W3C) guidelines for Web sites.

The online help for this product is also provided in HTML, which you can navigate using a screen reader.

Note You might need to configure your accessibility tool for optimal use. Some screen readers pronounce text based on its case; for example, they pronounce ALL UPPERCASE TEXT as initials, and MixedCase Text as words. You might find it helpful to configure your tool to announce syntax conventions. Consult the documentation for your tool.

For information about how Sybase supports accessibility, see Sybase Accessibility at http://www.sybase.com/accessibility. The Sybase Accessibility site includes links to information on Section 508 and W3C standards.

For a Section 508 compliance statement for Replication Server Options 15.1, see Sybase Accessibility at http://www.sybase.com/detail_list?id=52484.

If you need help

Each Sybase installation that has purchased a support contract has one or more designated people who are authorized to contact Sybase Technical Support. If you cannot resolve a problem using the manuals or online help, please have the designated person contact Sybase Technical Support or the Sybase subsidiary in your area.

CHAPTER 1

Setting Up Replication for Microsoft SQL Server

This chapter identifies the components that are required to implement a Sybase replication system for Microsoft SQL Server. This chapter contains an overview of the tasks and instructions for installing and configuring a sample replication environment for Microsoft SQL Server.

Note This procedure is only for proof-of-concept (POC) or testing, not for production.

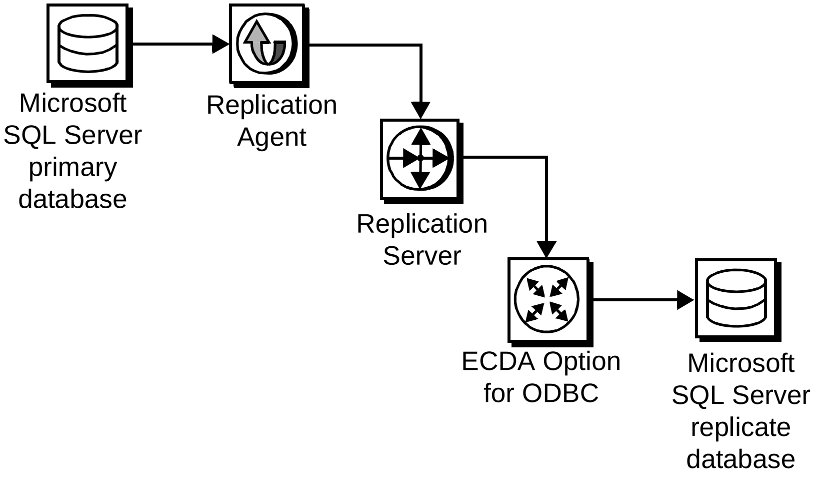
Sybase replication system

These components are required to implement a Sybase replication system for Microsoft SQL Server:

- A primary Microsoft SQL Server database
- A replicate Microsoft SQL Server database
- A Replication Server
- The Replication Server Options components:
    - Replication Agent
    - ECDA Option for ODBC

Figure 1-1 illustrates a Sybase replication system with Microsoft SQL Server databases. It shows the flow of data between the databases and the Replication Agent, Replication Server, and ECDA database gateway.

Figure 1-1: Sybase replication system for Microsoft SQL Server


Microsoft SQL Server primary database
Replication Agent
Replication Server
ECDA Option for ODBC
Microsoft SQL Server replicate database

Prerequisites

Before you begin setting up the replication system, you should be familiar with the Microsoft SQL Server data server and have an understanding of Sybase replication. In addition, these conditions must be met:

- ECDA for ODBC is installed on the same host as your replicate Microsoft SQL Server database.
- Replication Agent is installed on the same host as your primary Microsoft SQL Server database.
- You have obtained the licenses for the Replication Server and Replication Server Options.
- Microsoft SQL Server 2005 SP2, with database compatibility set to 2005(90), is installed and configured to serve as the source (primary) database and as the target (replicate) database from which Replication Agent replicates transactions.
- Microsoft SQL Server JDBC driver 1.2 is installed.

- TCP/IP connectivity is available.

During this procedure:

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

Note On Windows Vista, you must be logged in as an Administrator.

- Sybase recommends that you do not use Replication Server reserved words for object names and connection names. A complete list of reserved words are documented in Chapter 2 of the Replication Server *Reference Manual*.

System requirements

The target Windows computer on which you are installing the replication components must meet the following memory and disk space requirements:

Table 1-1: Memory and disk requirements

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

Installing replication components

The following subsections describe how to install the replication components and how to verify their installation.

Note Sybase recommends that you install all replication components on the same host where a Microsoft SQL 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.

Identifying the Sybase installation directory

Replication Agent, Replication Server, and ECDA are installed in the same base directory, which is identified by the SYBASE environment variable (*%SYBASE%*). Select a path on a host drive to be the recipient of the Sybase installation and configuration activities. For example:

```
c:\software\sybase
```

Accessing products from the SPDC Web site

If you are not using installation CDs, you must go to the Sybase product Download Center (SPCD) Web site to obtain replication software.

❖ **To download the replication software**

Note This procedure requires you repeat steps to download two products: Replication Server and Replication Server Options.

You must have a Login ID and Password to download software from the SPDC Web site.

1 Go to SPDC at https://sybase.subscribenet.com/control/sybs/login and enter your Login ID and Password.

The Product List page displays.

2 Select Replication Server.

The high-level Product Information page displays.

3. Select Replication Server again.

   The detailed Product Information page displays.

4. Select the Replication Server software you need by primary database, version, and platform (return later to this detailed Product Information page to select Replication Server Options software).

   The Software Terms and Conditions page displays, listing countries and regions.

5. Either:

   - Select I Agree to accept the terms without reading them, 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 displays. 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 go back to the high-level Product Information page.

6. The Product Download Page lists the software and documentation for the product. Select FTP Download 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 SyBooks download contains the rest of the documentation set.

7. Go back to the detailed Product Information page that you accessed in Step 3 and select the Replication Server Options software by primary database, version, and platform.

   The Software Terms and Conditions page displays, listing countries and regions.

8. Navigate the Software Terms and Conditions pages as you did in Step 5.

9. At the Product Download Page, select FTP Download for the three components of Replication Server Options to download and save the compressed files to your local drive:

   - Replication Agents

- ECDA Options
- DirectConnect Client

10 Uncompress and extract all the installation images to your local drive.

Installing Replication Agent

This section describes the steps for installing Replication Agent.

Note If Replication Agent is already installed, go to the next section called "Installing Replication Server" on page 9.

Installation guidelines

While installing Replication Agent, use the following guidelines:

- Install Replication Agent on a server where it can directly access the Microsoft SQL Server online transaction logs.
- Create a Replication Agent instance and configure it for the Microsoft SQL Server database from which you want to replicate.

Installing Replication Agent

Installing the Replication Agent is a process that creates a directory structure containing the Replication Agent binaries, scripts, JRE, and associated files. There are no configuration steps during installation.

❖ **To install Replication Agent software**

1 Use *one* of the following methods to install the software.

- After following the download instructions from the SPDC Web site in the section "Accessing products from the SPDC Web site" on page 4, go to the location where you extracted the Replication Agent 15.1 for Windows installation image, or
- Insert the Replication Agent CD into your CD-ROM drive.

2 Launch the installation program by executing the setup routine for Windows:

```
setup
```

3 If you see an error message saying that there is not enough temporary disk space, add the following to your setup command:

```
setup –is:tempdir someDirectory
```

Here, *someDirectory* is a directory with at least 100MB of disk space that the installation program can use.

4 On the Welcome window, click Next.

5 Select a geographic location that displays the license agreement. Select I Agree and click Next.

6 Enter the directory where you want to install Replication Agent:

- If the directory does not exist, the installation program prompts you to create it. Click Yes.
- If the destination directory exists, you receive a warning message that you are installing into an existing directory. Click Yes.

7 Select Typical for the installation setup type and click Next.

8 The installation program displays a summary of what is to be installed. Click Next.

9 The installation program starts to install Replication Agent, displaying the progress status. If you see the following message;

```
There are newer files already installed and do you
want to replace them?
```

Click No to All.

10 When you receive a message that the installation was successful, click Next.

11 The SySAM License Server window opens and displays this prompt:

```
Will licenses be obtained from the License Server?
```

Respond based on how you set up your environment and use the license obtained from the Software Product Download Center (SPDC) Web site for Replication Server Options and click Next

12 The SySAM notification window prompts you to configure your server for e-mail notification. When configuration is enabled, you receive information about license management events that require attention. At the SySAM notification window, do one of the following:

- Select Yes and either accept the default values that are supplied, or enter values for the following:

    - SMTP server host name
    - SMTP server port number
    - Sender e-mail
    - Recipient e-mail
    - Message severity level of an event that triggers e-mail. Your choices are:
        - Informational
        - Warning
        - Error
- If you choose not to have e-mail alerts or severity messages logged, select No.

Click Next.

13 When you receive a message indicating that Replication Agent was successfully installed, click Finish.

For more information for the replication installation process, see the Replication Agent *Installation Guide*.

Verifying the installation

❖ To verify the installation

1 In a command window, change to the directory where you installed Replication Agent.

2 Change the directory to *%SYBASE%\RAX-15_1\bin*.

3 Obtain the Replication Agent version string:

```
ra -v
```

If the Replication Agent installation completed successfully, you can see the Sybase copyright and the Replication Agent version string.

Installing Replication Server

This section describes the steps for a Replication Server installation and a procedure to verify the installation.

Note If Replication Server is already installed, go to the next section called "Installing the ECDA Option for ODBC" on page 12.

Installation guidelines

While installing Replication Server, use the following guidelines:

- Allocate a disk partition of at least 20MB for each Replication Server you are installing. You can add more partitions later, if necessary. Check each partition to make sure it 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.

Installing Replication Server

❖ **To install Replication Server**

1. Use *one* of the following methods to download Replication Server software.

   - After following the download instructions from the SPDC Web site in the section "Accessing products from the SPDC Web site" on page 4, go to the location where you extracted the Replication Server 15.0 for Windows installation image, or

   - Insert the Replication Server CD in your CD-ROM drive.

2. Launch the installation program by executing the setup routine for Solaris:

   ```
   setup
   ```

3. If you see an error message saying that there is not enough temporary disk space, add the following to your setup command:

   ```
   setup –is:tempdir someDirectory
   ```

   Here, *someDirectory* is a directory with at least 100MB of disk space that the installation program can use.

4. On the Welcome window, click Next.

5 Select a geographic location that displays the license agreement. Select I Agree and click Next.

6 Enter the directory where you want to install Replication Server:

- If the directory does not exist, the installation program prompts you to create it. Click Yes.
- If the destination directory exists, you receive a warning message that you are installing into an existing directory. Click Yes.

7 Select Typical for the installation setup type and click Next.

8 The installation program displays a summary of what is to be installed. Click Next.

9 The installation program starts to install Replication Server, displaying the progress status. You may see the following message:

```
There are newer files already installed and do you
want to replace them?
```

Click No to All.

10 When you receive a message indicating that Replication Server software has successfully installed, click Next.

Note This procedure uses the SAMPLE_RS Replication Server.

---

11 The SySAM License Server window opens and displays this prompt:

```
Will licenses be obtained from the License Server?
```

Respond based on how you set up your environment and use the license obtained from the Software Product Download (SPDC) Web site for Replication Server.

Click Next.

12 The SySAM notification window prompts you to 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 the following (from your worksheet):

- SMTP server host name
- SMTP server port number
- Sender e-mail

- Recipient e-mail
- Message severity level of an event that triggers e-mail. Your choices are:
    - Informational
    - Warning
    - Error

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

Click Next.

13 To start a sample Replication Server, select Yes and click Next.

14 This information is displayed and should be recorded:

```
This page contains detailed information regarding
the sample Replication Server. Please record this
information.

The sample Replication Server will be called
SAMPLE_RS and will run on port 11752. It will be
configured with a user of sa and no password.

The sample Replication Server will use an embedded
RSSD called SAMPLE_RS_ERSSD that runs on port 11751.
It will be configured with a user of
SAMPLE_RS_RSSD_prim and a password of
SAMPLE_RS_RSSD_prim_ps.

The installer has updated the appropriate interfaces
file or sql.ini file.

All files and logs associated with the sample
Replication Server will be located in the directory
%SYBASE%\REP-15_0\samp_repserver.

The sample Replication Server will be configured
using the file %SYBASE%\REP-15_0\samp_repserver
\SAMPLE_RS.res.
```

Click Next.

15 When the SAMPLE_RS is running and an “Installation was successful” message appears, click Finish.

Verifying the installation

❖ **To verify the installation**

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

2. Use the isql utility to log in to Replication Server:

   ```
   isql -Usa -P -SSAMPLE_RS
   ```

3. Exit the isql session with the quit command.

Installing the ECDA Option for ODBC

This section describes how to install the ECDA Option for ODBC.

Note If ECDA Option for ODBC is already installed, go to the next section called "Configuring replication components" on page 14.

Installing ECDA Option for ODBC

When installing ECDA for ODBC, be sure to install ECDA on the same server as the replicate database. This eliminates a network hop and improves performance.

❖ **To install ECDA for ODBC in GUI mode**

Note If you want to install using the console mode or a response file, refer to the ECDA *Installation Guide* for Microsoft Windows.

1. Use *one* of the following methods to download the software:

   - After following the download instructions from the SPDC Web site in the section "Accessing products from the SPDC Web site" on page 4, go to the location where you extracted the ECDA Option for ODBC 15.0 installation image.

   - Insert the CD labeled "ECDA 15.0 Option for ODBC for Windows."

2. Launch InstallShield by executing the Windows setup routine:

   ```
   setup
   ```

3 If you see an error message saying that there is not enough temporary disk space, add the following to your setup command:

```
setup –is:tempdir someDirectory
```

Here, *someDirectory* is a directory with at least 100MB of disk space that the installation program can use.

4 On the Welcome window, click Next.

5 Select a geographic location that displays the license agreement. Select I Agree and click Next.

6 Enter the directory where you want to install ECDA:

- In the destination directory, enter the directory where the Replication Agent software is installed. Click Yes to install into an existing directory.
- If the destination directory exists, you receive a warning message that you are installing into an existing directory. Click Yes.

7 Select Custom setup, and click Next.

8 Deselect ECDA Option for Oracle, and click Next.

9 Verify the modules to install. Click Next.

You may receive this message:

```
There are newer files already installed and do you
want to replace them
```

Click No to All.

10 When you receive a message that the installation was successful, click Next.

11 The SySAM License Server window opens and displays this prompt:

```
Will licenses be obtained from the License Server?
```

Respond based on how you set up your environment and use the license obtained from the Software Product Download Center (SPDC) Web site for Replication Server Options.

Click Next.

12 The SySAM notification window prompts you to 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 the following (from your worksheet):

    - SMTP server host name
    - SMTP server port number
    - Sender e-mail
    - Recipient e-mail
    - Message severity level of an event that triggers e-mail. Your choices are:
        - Informational
        - Warning
        - Error

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

13 When you receive a message indicating that ECDA was successfully installed, click Finish.

ECDA for ODBC, including the DirectConnect server, is now installed.

Configuring replication components

This section describes the tasks you must perform to configure each of the components in the replication system.

Configuring ECDA Option for ODBC

Configuring the ECDA Option for ODBC involves the following sequence of tasks:

1. Configuring the Microsoft SQL Server ODBC driver
2. Configuring and starting the DirectConnect server

3 Creating a maintenance user for replication

4 Creating the replicate objects

5 Verifying the ECDA installed objectss

Configuring the Microsoft SQL Server ODBC driver

If you already have a data source configured for your replicate database, skip this section and continue with “Configuring and starting the DirectConnect server” on page 16.

When you installed Microsoft SQL Server, an ODBC driver was also installed. Configure this driver to connect to the replicate database.

❖ To configure the Microsoft SQL Server ODBC driver

1 Open the ODBC Driver Manager by navigating to Control Panel | Administrative Tools and double-clicking the Data Sources (ODBC) icon.

2 Click on 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 on the System DSN tab, click Add, 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 Check the Change the Default Database To check box, 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.

Configuring and starting the DirectConnect server

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

- 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.

❖ Creating and configuring a DirectConnect server

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 window 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 Using a text editor, open the *%SYBASE%\DC-15_0\servername\cfg\dcany.cfg* file, where *servername* is the name of the server you created in step 3 using the DCWizard.

7 In the file, locate the name of the Microsoft SQL Server access service that you created in step 4 using the DCWizard. Under that, set the ConnectionSpec1 parameter to the name of the Microsoft SQL Server ODBC data source you created. For example:

```
[Service Library]
{Logging}
LogSvcLibStatistics=0
{Client Interaction}
SvclibDescription=Access Service Library for ODBC
[my_mssql_access_service]
{ACS Required}
ConnectionSpec1=my_mssql_datasource_name
```

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 the *DC_SYBASE.bat* script 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
```

Here, *servername* is the name of the DirectConnect server.

9 Click Finish.

❖ To verify that you can connect to Microsoft SQL Server

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 sourcing the *%SYBASE%\DC_SYBASE.bat* file.

4 Use the Sybase isql utility to log in to Microsoft SQL Server through the DirectConnect server:

```
isql –Uvaliduser –Ppassword –SDCServer
```

Here, *validuser* and *password* are any valid user and password, and *DCServer* is the DirectConnect server name identified in the previous procedure.

5 Verify the connection to the replicate Microsoft SQL Server database by obtaining the DBMS name and version number:

```
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

This section explains how to create a maintenance user in the replicate Microsoft SQL Server database.

Description of a maintenance user

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. Replication Server requires one maintenance user to be defined for each connection.

❖ **To create a maintenance user in Microsoft SQL Server**

1 Using the Microsoft SQL Server Management Studio or the sqlcmd utility, connect to the replicate database, and enter the following command to create maintenance user login:

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

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

2 At the replicate database, enter the following 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 the replicate objects

❖ **To create the replicate objects**

- Sign on to the Microsoft SQL Server database through ECDA Option for ODBC as the maintenance user (assuming that the maintenance user that you created has the resource role to use create table, and execute this script using the Sybase isql utility):

```
isql –Umaintuser –Ppassword –Smy_mssql_access_service
-i %SYBASE%\REP-15_0\scripts\hds_msss_setup_for_replicate.sql
```

Here, *my_mssql_access_service* is the name of the DirectConnect access service.

Verifying the ECDA installed objects

❖ **To verify the ECDA installed objects**

1. Open another command window and navigate to the DirectConnect installation. For example:

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

2. Set the environment variables by executing the *DC_SYBASE.bat* file.

3. Use the Sybase isql utility to log in to the replicate Microsoft SQL Server database through ECDA as *maintuser*.

- To verify whether the rs_info table has been created successfully:

```
isql –Umaintuser –Ppassword -Smy_mssql_access_service
select * from rs_info
go
```

This following result is returned:

```
RSKEY            RSVAL
-------------------- ----------------

charset_name     cp850
sortorder_name   bin_cp850
(2 rows affected)
```

- To verify whether the rs_lastcommit table has been created successfully:

```
isql –Umaintuser –Ppassword –Smy_mssql_access_service
select * from rs_lastcommit
go
```

The following is 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 are displayed, and you encounter an error message.

4 Exit the isql session with the quit command.

Configuring Replication Server

Configuring Replication Server includes the following tasks:

- Configuring Replication Server for Microsoft SQL Server replication
- Configuring Replication Server for replication to the target database
- Configuring Replication Server for replication from the primary database

Configuring Replication Server for Microsoft SQL Server replication

This section describes the configuration tasks for configuring Replication Server for Microsoft SQL Server.

❖ **To apply heterogeneous datatype support (HDS) scripts to RSSD**

Prepare the Replication Server RSSD for replication to a Microsoft SQL Server database. To apply HDS support scripts to RSSD:

1 Set the environment variables by executing the *SYBASE.bat* file.

2 Change to the *%SYBASE%\REP-15_0\scripts* directory:

3 Load the following scripts into the RSSD:

```
isql -U SAMPLE_RS_RSSD_prim -P SAMPLE_RS_RSSD_prim_ps –SSAMPLE_RS_ERSSD
–ihds_clt_ase_to_msss.sql -DSAMPLE_RS_ERSSD

isql -U SAMPLE_RS_RSSD_prim -P SAMPLE_RS_RSSD_prim_ps –SSAMPLE_RS_ERSSD
–ihds_msss_udds.sql -DSAMPLE_RS_ERSSD

isql -U SAMPLE_RS_RSSD_prim -P SAMPLE_RS_RSSD_prim_ps –SSAMPLE_RS_ERSSD
–ihds_msss_funcstrings.sql -DSAMPLE_RS_ERSSD
```

4 Shut down Replication Server:

```
isql -Usa -P -SSAMPLE_RS
shutdown
go
```

5 Navigate to the *%SYBASE%\REP-15_0\samp_repserver* directory:

6 Run the *RUN_SAMPLE_RS* script, to start the SAMPLE_RS Replication Server.

Configuring Replication Server for replication to the target database

This section describes the required configurations for Replication Server for replication to the target Microsoft SQL Server database.

❖ To create a Replication Server connection to replicate

1 Navigate to the Replication Agent scripts directory: *%SYBASE%\REP-15_0\scripts*.

2 Make a copy of the *hds_msss_connection_sample.sql* script:

```
cp hds_msss_connection_sample.sql
my_mssql_create_rs_repl_connection.sql
```

3 Before executing the *my_mssql_create_rs_repl_connection.sql* script against your Replication Server, change the *msss_dco.dbname* to the name of the connection that Replication Server uses to connect to the DirectConnect access service, where:

- *msss_dco* is the DirectConnect access service name.
- *dbname* is any valid identifier. Sybase 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" on page 18. For example:

```
create connection to msss_dco.dbname
set error class rs_sqlserver_error_class
set function string class rs_msss_function_class
set username maintuser
set password "password"
set batch to "off"
with log transfer on
```

```
    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 using the Sybase isql utility:

```
isql -Usa -P -SSAMPLE_RS -i my_mssql_create_rs_repl_connection.sql
```

5 To verify the Replication Server connection to the replicate database, log in to SAMPLE_RS and verify the Replication Server connection to the replicate database:

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

You should see a message returned for the DSI connection *msss_dco.dbname*. 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

This section describes the configuration steps required for replication from the primary database. It includes:

- Creating a Replication Server connection to the primary database
- Creating a database replication definition
- Creating a database replication subscription

❖ To create a Replication Server connection to the primary database

1 Go to the Replication Agent scripts directory, *%SYBASE%\RAX-15_1\scripts\sybase*

2 Make a copy of the *rs_create_test_primary_connection.sql* script:

```
cp rs_create_test_primary_connection.sql
my_mssql_rs_create_pri_connection.sql
```

3 Before executing the *my_mssql_rs_create_pri_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 rs_source_ds (located in *%SYBASE%\RAX-15_1\init\mssql.rs*).

- *test* is rs_source_db (located in *%SYBASE%\RAX-15_1\init\mssql.rs*).

Note You must save the defined values to be used for the Replication Agent *mssql.rs* resource file later in the process.

For example, *NY.NYmss5*.

4 Execute the script in Replication Server:

```
isql -Usa -P –SSAMPLE_RS -i my_mssql_rs_create_primary_connection.sql
```

A message is displayed that indicates the Replication Server connection to the primary database is created.

❖ To create a database replication definition

1 Navigate to the *%SBYBASE%\RAX_15-1\scripts\sybase* directory.

2 Make a copy of the *rs_create_test_db_repdef.sql* script:

```
cp rs_create_test_db_repdef.sql
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 rs_source_ds, located in *%SYBASE%\RAX-15_1\init\mssql.rs*.
- *pdb* is rs_source_db, located in *%SYBASE%\RAX-15_1\init\mssql.rs*.

For example, *NY.NYmss5*.

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 -P –SSAMPLE_RS -i my_mssql_rs_create_test_db_repdef.sql
```

A message is displayed that indicates the database replication definition is created.

❖ To create a database replication subscription

1 Make a copy of the *rs_create_test_db_sub.sql* script:

```
cp rs_create_test_db_sub.sql
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 rs_source_ds, located in *%SYBASE%\RAX-15_1\init\mssql.rs*.
- *pdb* is rs_source_db, located in *%SYBASE%\RAX-15_1\init\mssql.rs*.
- *rds* is the DirectConnect access service name.
- *rdb* is the name of the replicate database.

Here is an example of create 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 Replication Server and execute:

```
isql -Usa -P –SSAMPLE_RS -i my_mssql_rs_create_test_db_sub.sql
```

A message is displayed that indicates the subscription is in the process of being created.

Configuring Replication Agent

Configuring Replication Agent involves the following sequence of tasks:

Note On Windows Vista, you must be logged in as an Administrator.

1 Configuring and verifying the primary Microsoft SQL Server

2 Configuring and validating the Replication Agent resource file

3 First-time initialization

4 Subsequent initialization

5 Verifying the Replication Agent Replicating state

Configuring and verifying the primary Microsoft SQL Server

Before you configure Replication Agent and configure the primary Microsoft SQL Server database for replication, complete these procedures, which are explained in the following subsections.

- Verify the compatibility level of Microsoft SQL Server.
- Disable Microsoft SQL Server replication.
- Create a Microsoft SQL Server user and grant Microsoft SQL Server permissions.
- Enable remote DAC.
- Obtain the Microsoft SQL Server DAC port number.
- Verify the supplemental logging of primary key data.
- Make the primary transaction log files readable for Replication Agent.
- Stop the Analysis Service.
- Enable TCP/IP.

❖ To verify the compatibility level of Microsoft SQL Server

Replication Agent supports only Microsoft SQL Server 2005 Service Pack 2 and later. Verify that the database compatibility level is set to Microsoft SQL Server 2005 (90).

1. In the Microsoft SQL Server Management Studio, right-click on your primary database and select Properties. This brings up the Database Properties dialog box.
2. In Database Properties, click Options, and verify that the Compatibility level is “Microsoft SQL Server 2005 (90).” If it is not, select that compatibility from the drop-down list and click OK.

❖ To disable 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, you must remove the publication before using Replication Agent for Microsoft SQL Server.

❖ To create a Microsoft SQL Server user and grant permissions

- Using Microsoft isql, log in to the primary Microsoft SQL Server as a system administrator, and run the following commands to create a Microsoft SQL Server user named "ra_user" with the password "sybase," and grant permissions to the user:

```
use master;
create login ra_user;

use <primary_database>;
create user ra_user for login ra_user;

EXEC sp_addsrvrolemember 'ra_user','sysadmin';
```

❖ To enable remote DAC

1 From the Windows Start menu, choose Microsoft SQL Server | Configuration Tools | Surface Area Configuration | Surface Area Configuration for Features.

2 In the Surface Area Configuration for Features window, choose DAC under the MSSQLSERVER/Database Engine tree, and make sure the Enable remote DAC check box is selected.

❖ To obtain the Microsoft SQL Server DAC port number

1 Open the *ERRORLOG* file in a text editor. 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 the string "Dedicated admin" to find an entry similar to the following:

```
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.

❖ To verify the version of the Microsoft Filer Manager Library

To make the primary transaction log files readable for Replication Agent, the Microsoft Filter Manager Library must be version 5.1.2600.2978 or later.

1 To determine the version of the library, in Windows Explorer, right-click *c:\windows\system32\fltlib.dll*, select Properties, and click the Version tab in the Properties dialog box.

2 If the version is earlier than 5.1.2600.2978, go to the Microsoft Web site at *http://windowsupdate.microsoft.com*, and update your Windows system.

❖ To make the primary transaction log files readable for Replication Agent

Perform the following steps to 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. On Windows, this directory is located at *%SYBASE%\RAX-15_1\system\<platform>*. Here, *<platform>* is either *winx86* or *winx64*.

2 Right-click the *sybfilter.inf* file to install the *sybfilter* driver.

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_1\system\<platform>*, 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 the your configuration file.

5 In Windows Explorer, navigate to *%SYBASE%\RAX-15_1\bin*, and double-click the s*ybfiltermgr.exe* file 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.

- *User manager* - Use the add command in the management console. The syntax for this command is as follows:

  add *serverName dbName logFilePath*

For example, to add the log file named *pdb2_log.ldf* at *C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\Data\* to the dbName database on the *serverName* data server, use the following:

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

Note If you add the log file path with the user manager, 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
```

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

8. If you added a log file for your primary database before adding the log file path to the *sybfilter* driver, you must 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.

For more information on the *sybfilter* driver, see the Replication Agent *Primary Database Guide*.

❖ To stop the Analysis Service

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

❖ **To enable TCP/IP**

1 From the Windows Start menu, choose Microsoft SQL Server | Surface Area Configuration | Surface Area Configuration for Services and Connections.

2 Under the Database Engine tree, click Remote Connections.

3 Select “Local and remote connections,” and “Using both TCP/IP and Named Pipes protocols.“

❖ **To install the Microsoft SQL Server JDBC driver**

1 Go to the Microsoft download site (*http://www.microsoft.com/downloads)* and search for the following string:

```
Microsoft SQL Server 2005 JDBC driver
```

2 Click on the link for the 1.2 driver, and follow the instructions to download and install the driver.

3 Open Control Panel | System, click on 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 using the semi-colon (;) 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. This section describes how to prepare for and create a Replication Agent instance.

❖ **To configure and validate the Replication Agent parameters in a resource file**

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 it provides a means to record or retain the configuration information for a Replication Agent instance, allowing an instance to be removed and recreated.

The Microsoft SQL Server resource file template can be found at *%SYBASE%\RAX-15_1\init\mssql.rs*.

2 Create an instance resource file.

Copy the resource file template *%SYBASE%\RAX-15_1\init\mssql.rs* to another file. For example:

```
cp mssql.rs myra.rs
```

Using a text editor, 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.

The resource file is self-documenting, with each parameter described. For more information, refer to the Replication Agent *Administration Guide*, in Chapter 2, “Setup and Configuration,” the section titled, “Creating a Replication Agent instance using resource files.”

Table 1-2: 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 capabilities.	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 Replication Server's configuration file, *%SYBASE%\REP-15_0 \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.	SAMPLE_RS_RSSD_maint_ps
start_instance	Automatically start the instance that was created.	yes
initialize_instance	Automatically initialize the new Replication Agent instance.	yes

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.

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_1\bin\ra_admin -vr myra.rs
```

Validation results are returned as one of the following:

- `Response-file processing completed`
- `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 because of this process.

First-time 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. The data server-level modifications are required only once. Use this procedure if you are initializing the primary data server and a Replication Agent instance for the first time.

If you have previously initialized the primary data server and a Replication Agent instance and want to initialize another Replication Agent instance, skip this procedure, and follow the instructions in "Subsequent initialization" on page 34.

❖ To 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
```

Here, *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 the resource file has been validated, use the ra_admin utility to create the Replication Agent instance using the -r parameter. For example:

```
%SYBASE%\RAX-15_1\bin\ra_admin -r myra.rs
```

Execution results are returned as one of the following:

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

For more information about the resource file or its use, refer to the Replication Agent *Administration Guide*, Chapter 2, “Setup and Configuration,” in the section titled, “Creating a Replication Agent instance using resource files.”

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_1\myra
```

Execute the *RUN* file, for example:

```
RUN_myra
```

5 Using the dsedit utility, update the Replication Server *sql.ini* file with the Replication Agent location, and verify the connection to the 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 Use the Sybase isql utility to log in to Replication Agent:

```
isql –Usa –P –Smyra
```

Here, *myra* is the name of Replication Agent.

6 If your resource file set initialize_instance 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 the following command:

```
pdb_xlog init
```

7 Stop the Microsoft SQL Server in single-user mode:

a Use the sqlcmd utility to log in to the server:

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

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

b Issue the shutdown command.

8 Restart Microsoft SQL Server in multi-user 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.

The Replication Agent instance is now running, and the primary data server and the Replication Agent instance have been initialized. Continue with "Verifying the Replication Agent Replicating state" on page 35.

Subsequent initialization

Use the following procedure only if you have already initialized the primary data server, as described in "First-time initialization" on page 32, and you want to create another Replication Agent instance. This procedure creates and initializes only the Replication Agent instance, not the primary data server.

❖ To create and initialize a Replication Agent instance

1. Execute the resource file to create the Replication Agent instance.

   - After the resource file has been validated as described in "Configuring and validating the Replication Agent resource file" on page 29, use the ra_admin utility to create the Replication Agent instance using the -r parameter, for example:

     ```
     %SYBASE%\RAX-15_1\bin\ra_admin -r myra.rs
     ```

     Execution results are returned as one of the following:

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

     For more information about the resource file or its use, refer to the Replication Agent *Administration Guide,* Chapter 2, "Setup and Configuration," in the section titled, "Creating a Replication Agent instance using resource files."

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

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

   b. To start the Replication Agent instance, execute the *RUN* file, for example:

```
    RUN_myra
```

3 Using the dsedit utility, update the Replication Server *sql.ini* file with the Replication Agent location, and verify the connection to the 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 Use the Sybase isql utility to log in to Replication Agent:

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

   Here, *myra* is the name of Replication Agent.

4 If your resource file set initialize_instance 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 the following Replication Agent command:

```
pdb_xlog init
```

Your Replication Agent instance is now running and has been initialized. Continue with "Verifying the Replication Agent Replicating state" on page 35.

Verifying the Replication Agent *Replicating* state

❖ **To verify the Replication Agent *Replicating* state**

1 Enter:

```
resume
go
```

If the Replication Agent successfully transfers to a replicating state, the following result is returned:

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

A returned state of *Admin* indicates that an error prevented the replicating state from being achieved. To determine the error, review the contents of the Replication Agent system log.

2 The ra_status command returns the state of the Replication Agent. It is good practice to 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.
```

A returned state of *Admin* indicates that an error prevented the *Replicating* state from being achieved. To determine the error, review the contents of the Replication Agent system log.

3 Validate that both primary and replicate connections are active:

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

Note Be aware that:

- The DSI connection for the primary database connection is usually down 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 has similar status for threads to the following:

```
admin who
go
```

The following is displayed:

```
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)
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) NY.NYmss5

DSI Suspended 103 NY.NYmss5

24 DIST Awaiting Wakeup 103 NY.NYmss5

25 SQT Awaiting Wakeup 103:1 DIST NY.NYmss5

23 SQM Awaiting Message 103:1 NY.NYmss5

22 SQM Awaiting Message 103:0 NY.NYmss5

62 REP AGENT Awaiting Command NY.NYmss5
```

Testing replication

After you finish setting up the replication system, you can test replication.

❖ **To test replication**

1. Connect to the primary Microsoft SQL Server database as a regular user, not 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
   go
   ```

4. Connect to the replicate Microsoft SQL Server database as a regular user, not the maintenance user.

5 Create a test table in the replicate database:

```
create table T1 (a int b chart(10))
go
```

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

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

7 Connect to the Replication Agent and run the following command to 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. See “Configuring Replication Server for replication from the primary database” on page 22.

Replication Agent applies only committed transactions to a replicate database.

```
insert into T1 values ( 42, ‘foo’ )
go
```

9 Use your preferred Microsoft SQL Server 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 Microsoft SQL Server database for replication

In a test environment, you may occasionally need to reset the replication environment. Instead of deleting and re-creating a new Replication Agent instance, use this procedure to facilitate resetting the environment.

❖ **To reset the primary database for replication**

1 Protect your new environment from old log information by using the following command in the Replication Agent to:

```
pdb_xlog move_truncpt
```

2 To retain marking and log device information, reinitialize the Replication Agent using pdb_xlog init with the force option, which forces the Replication Agent repository to be refreshed instead of overwritten:

```
pdb_xlog init, force
```

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

3 Zero-out the locator stored in Replication Server using the rs_zeroltm command. For example:

```
isql -USAMPLE_RS_RSSD_prim -PSAMPLE_RS_RSSD_prim_ps
-SSAMPLE_RS_ERSSD -DSAMPLE_RS_ERSSD


rs_zeroltm NY, NYmss5
go
```

The following is displayed:

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

CHAPTER 2 Setting Up Replication for Oracle

This chapter identifies the components that are required to implement a Sybase replication system for Oracle. This chapter contains an overview of the tasks and instructions for installing and configuring a sample replication environment for Oracle.

Note This procedure is only for proof-of-concept (POC) or testing, not for production.

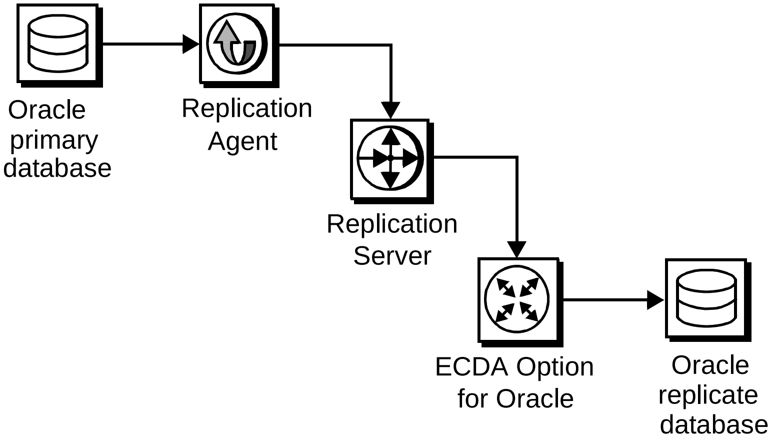
Sybase replication system

These components are required to implement a Sybase replication system for Oracle:

- A primary Oracle data server
- A replicate Oracle data server
- A Replication Server
- The Replication Server Options components:
    - Replication Agent
    - ECDA Option for Oracle

Figure 2-1 illustrates a Sybase replication system with Oracle servers. It shows the flow of data between the data servers and the Replication Agent, Replication Server, and ECDA database gateway.

Figure 2-1: Sybase replication system for Oracle



Prerequisites

Before you begin setting up the replication system, you should be familiar with Oracle data servers and have an understanding of Sybase replication. In addition, these conditions must be met:

- You must have a valid license for Replication Server and Replication Server Options.
- Oracle Enterprise Edition 9.2 or 10.2, is installed and configured, to serve as the source (primary) database and as the target (replicate) database from which Replication Agent replicates transactions.
- Oracle JDBC thin driver for Oracle 10.2 and for JDK 1.4 and 1.5 is installed.
- TCP/IP connectivity is available.
- Any OS patches required for Java 5.0 have been installed.
- The Replication Agent must have direct access to the Oracle online and archived *redo* logs.

During this procedure:

- Because some directories, files, executable commands, and examples are shown only for Sun Solaris, you must adjust accordingly for Windows and for other UNIX platforms.
- Sybase recommends that you do not use Replication Server reserved words for object names and connection names. A complete list of reserved words are documented in Chapter 2 of the Replication Server *Reference Manual*.

System requirements

The target computer on which you are installing the replication components must meet the following memory and disk space requirements:

For Windows

Table 2-1: Windows memory and disk requirements

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

For UNIX

Table 2-2: UNIX memory and disk requirements

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

For Linux

Table 2-3: Linux memory and disk requirements

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

Installing replication components

The following subsections describe how to install the replication components and how to verify their installation.

Note The replication of sequences is not covered in this document. If you need to set up sequence replication, refer to the Replication Agent *Installation Guide* and the Replication Agent *Primary Database Guide*.

Identifying the Sybase installation directory

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

```
/software/sybase
```

Accessing products from the SPDC Web site

If you are not using installation CDs, you must go to the Sybase product Download Center (SPCD) Web site to obtain replication software.

❖ **To download the replication software**

Note This procedure requires you repeat steps to download two products: Replication Server and Replication Server Options.

You must have a Login ID and Password to download software from the SPDC Web site.

1 Go to SPDC at https://sybase.subscribenet.com/control/sybs/login and enter your Login ID and Password.

The Product List page displays.

2 Select Replication Server.

The high-level Product Information page displays.

3 Select Replication Server again.

The detailed Product Information page displays.

4 Select the Replication Server software you need by primary database, version, and platform (return later to this detailed Product Information page to select Replication Server Options software).

The Software Terms and Conditions page displays, listing countries and regions.

5 Either:

- Select I Agree to accept the terms without reading them, 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 displays. 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 go back to the high-level Product Information page.

6 The Product Download Page lists the software and documentation for the product. Select FTP Download 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 SyBooks download contains the rest of the documentation set.

7 Go back to the detailed Product Information page that you accessed in Step 3 and select the Replication Server Options software by primary database, version, and platform.

The Software Terms and Conditions page displays, listing countries and regions.

8 Navigate the Software Terms and Conditions pages as you did in Step 5.

9 At the Product Download Page, select FTP Download for the three components of Replication Server Options to download and save the compressed files to your local drive:

- Replication Agents
- ECDA Options

- DirectConnect Client

10 Uncompress and extract all the installation images to your local drive.

Installing Replication Agent

This section describes the steps for installing Replication Agent.

Note If Replication Agent is already installed, go to the next section called "Installing Replication Server" on page 49.

Installation guidelines

While installing Replication Agent, use the following guidelines:

- Install Replication Agent for Oracle on a server where it can directly access the Oracle online *redo* logs and the archived *redo* logs.
- Create a Replication Agent instance and configure it for the Oracle database from which you want to replicate.

Installing Replication Agent

Installing the Replication Agent is a process that creates a directory structure containing the Replication Agent binaries, scripts, JRE, and associated files. There are no configuration steps during installation.

❖ To install Replication Agent software

1 Use *one* of the following methods to install the software.

- After following the download instructions from the SPDC Web site in the section "Accessing products from the SPDC Web site" on page 44, go to the location where you extracted the Replication Agent 15.1 for Sun Solaris installation image, or
- Insert the Replication Agent CD into your CD drive.

2 Launch the installation program by executing the setup routine for Solaris:

```
./setup
```

3 If you see an error message saying that there is not enough temporary disk space, add the following to your setup command:

```
./setup -is:tempdir someDirectory
```

Here, *someDirectory* is a directory with at least 100MB of disk space that the installation program can use.

Note If you are installing from a remote machine and do not have an X-windows client, execute ./setup –console, to run in console mode.

4 On the Welcome window, click Next.

5 Select a geographic location that displays the license agreement. Select I Agree and click Next.

6 Enter the directory where you want to install Replication Agent:

- If the directory does not exist, the installation program prompts you to create it. Click Yes.
- If the destination directory exists, you receive a warning message that you are installing into an existing directory. Click Yes.

7 Select Typical for the installation setup type and click Next.

8 The installation program displays a summary of what is to be installed. Click Next.

9 The installation program starts to install Replication Agent, displaying the progress status. If you see the following message;

```
There are newer files already installed and do you
want to replace them?
```

Click No to All.

10 When you receive a message that the installation was successful, click Next.

11 The SySAM License Server window opens and displays this prompt:

```
Will licenses be obtained from the License Server?
```

Respond based on how you set up your environment and use the license obtained from the Software Product Download Center (SPDC) Web site for Replication Server Options and click Next.

12 The SySAM notification window prompts you to configure your server for e-mail notification. When configuration is enabled, you receive information about license management events that require attention. At the SySAM notification window, do one of the following:

- Select Yes and either accept the default values that are supplied, or enter values for the following:
  - SMTP server host name
  - SMTP server port number
  - Sender e-mail
  - Recipient e-mail
  - Message severity level of an event that triggers e-mail. Your choices are:
    - Informational
    - Warning
    - Error
- If you choose not to have e-mail alerts or severity messages logged, select No.

Click Next.

13 When you receive a message indicating that Replication Agent was successfully installed, click Finish.

For more information for the replication installation process, see the Replication Agent *Installation Guide*.

Verifying the installation

❖ To verify the installation

1. Change to the directory where you installed Replication Agent.
2. Set the environment variable by sourcing the *SYBASE.csh* file.
3. Change the directory to *$SYBASE/RAX-15_1/bin*.
4. Obtain the Replication Agent version string:

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

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

Installing Replication Server

This section describes the steps for a Replication Server installation and a procedure to verify the installation.

Note If Replication Server is already installed go to the next section called "Installing the ECDA Option for Oracle" on page 52.

Installation guidelines

While installing Replication Server, use the following guidelines:

- Allocate a disk partition of at least 20MB for each Replication Server you are installing. You can add more partitions later, if necessary. Check each partition to make sure it 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.

Installing Replication Server

❖ To install Replication Server

1 Use *one* of the following methods to download Replication Server software.

- After following the download instructions from the SPDC Web site in the section "Accessing products from the SPDC Web site" on page 44, go to the location where you extracted the Replication Server 15.0 for Sun Solaris installation image, or
- Insert the Replication Server CD in your CD drive.

2 Launch the installation program by executing the setup routine for Solaris:

```
./setup
```

3 If you see an error message saying that there is not enough temporary disk space, add the following to your setup command:

```
./setup –is:tempdir someDirectory
```

Here, *someDirectory* is a directory with at least 100MB of disk space that the installation program can use.

4 On the Welcome window, click Next.

5 Select a geographic location that displays the license agreement. Select I agree and click Next.

6 Enter the directory where you want to install Replication Server:

- If the directory does not exist, the installation program prompts you to create it. Click Yes.
- If the destination directory exists, you receive a warning message that you are installing into an existing directory. Click Yes.

7 Select Typical for the installation setup type and click Next.

8 The installation program displays a summary of what is to be installed. Click Next.

9 The installation program starts to install Replication Server, displaying the progress status. You may see the following message,

```
There are newer files already installed and do you
want to replace them?
```

Click No to All.

10 When you receive a message indicating that Replication Server software has successfully installed, click Next.

Note This procedure uses the SAMPLE_RS Replication Server.

11 The SySAM License Server window opens and displays this prompt:

```
Will licenses be obtained from the License Server?
```

Respond based on how you set up your environment and use the license obtained from the Software Product Download (SPDC) Web site for Replication Server.

Click Next.

12 The SySAM notification window prompts you to 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 the following (from your worksheet):

- SMTP server host name
- SMTP server port number
- Sender e-mail

- Recipient e-mail
- Message severity level of an event that triggers e-mail. Your choices are:
  - Informational
  - Warning
  - Error

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

Click Next.

13. To start a sample Replication Server, select Yes and click Next.
14. This information is displayed and should be recorded:

```
This page contains detailed information regarding
the sample Replication Server. Please record this
information.

The sample Replication Server will be called
SAMPLE_RS and will run on port 11752. It will be
configured with a user of sa and no password.

The sample Replication Server will use an embedded
RSSD called SAMPLE_RS_ERSSD that runs on port 11751.
It will be configured with a user of
SAMPLE_RS_RSSD_prim and a password of
SAMPLE_RS_RSSD_prim_ps.

The installer has updated the appropriate interfaces
file or sql.ini file.

All files and logs associated with the sample
Replication Server will be located in the directory
$SYBASE/REP-15_0/samp_repserver.

The sample Replication Server will be configured
using the file $SYBASE/REP-15_0/samp_repserver
/SAMPLE_RS.res.
```

Click Next.

15. When the SAMPLE_RS is running and an “Installation was successful” message appears, click Finish.

Verifying the installation

❖ **To verify the installation**

1. Change to the directory where you installed Replication Server.
2. Set the environment variables by sourcing the *SYBASE.csh* file.
3. Use the isql utility to log in to Replication Server:

   ```
   isql -Usa -P -SSAMPLE_RS
   ```
4. Exit the isql session with the quit command.

Installing the ECDA Option for Oracle

This section describes how to install the ECDA Option for Oracle.

Note If ECDA Option for Oracle was already installed, go to the next section called “Configuring replication components.”

Installing ECDA Option for Oracle

When installing ECDA for Oracle, be sure to install ECDA on the same server as the replicate database. This eliminates a network hop and improves performance.

❖ **To install ECDA for Oracle in GUI mode**

Note If you want to install using the console mode or a response file, refer to the ECDA *Installation Guide* for Linux and UNIX.

---

1. Use *one* of the following methods to download the software:
   - After following the download instructions from the SPDC Web site in the section “Accessing products from the SPDC Web site” on page 44, go to the location where you extracted the ECDA Option for Oracle 15.0 installation image.
   - Insert the CD labeled ECDA 15.0 Option for Oracle for Sun Solaris.
2. Launch InstallShield by executing the Sun Solaris setup routine:

   ```
   ./setup
   ```

3 If you see an error message saying that there is not enough temporary disk space, add the following to your setup command:

```
./setup –is:tempdir someDirectory
```

Here, *someDirectory* is a directory with at least 100MB of disk space that the installation program can use.

4 On the Welcome window, click Next.

5 Select a geographic location that displays the license agreement. Select I agree and click Next.

6 Enter the directory where you want to install ECDA:

- In the destination directory, enter the directory where the Replication Agent software is installed. Click Yes to install into an existing directory.
- If the destination directory exists, you receive a warning message that you are installing into an existing directory. Click Yes.

7 Select Custom setup, and click Next.

8 Deselect ECDA Option for ODBC, and click Next.

9 Verify the modules to install. Click Next.

You may receive a message:

```
There are newer files already installed and do you
want to replace them
```

Click No to All.

10 When you receive a message that the installation was successful, click Next.

11 The SySAM License Server window opens and displays this prompt:

```
Will licenses be obtained from the License Server?
```

Respond based on how you set up your environment and use the license obtained from the Software Product Download Center (SPDC) Web site for Replication Server Options.

Click Next.

12 The SySAM notification window prompts you to 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 the following (from your worksheet):

   - SMTP server host name
   - SMTP server port number
   - Sender e-mail
   - Recipient e-mail
   - Message severity level of an event that triggers e-mail. Your choices are:
     - Informational
     - Warning
     - Error

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

   Click Next.

13 When you receive a message indicating that ECDA was successfully installed, click Finish.

   ECDA, including the DirectConnect server, is installed.

Configuring replication components

This section describes the tasks you must perform to configure each of the replication component in the replication system.

Configuring ECDA Option for Oracle

Configuring ECDA Option for Oracle involves the following sequence of tasks:

1 Configuring and starting the DirectConnect server

2 Creating a maintenance user for replication

3 Creating objects and public synonyms

4 Verifying the ECDA installed objects

Configuring and starting the DirectConnect server

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

- You have made a copy of the existing *tnsnames.ora* file, currently being used to connect to Oracle, and have placed it in a temporary file location.
- The Oracle connect string is available.
- The name for a valid Oracle account, which is used as the administrator for the DirectConnect server, is available.
- The number of the unused port to be used by the DirectConnect server is available.

❖ Creating and configuring a DirectConnect server

1 Start the Create Server wizard navigate to the *$SYBASE/DCO-15_0/DCWizard* directory and execute *DCWizard.sh* script

The Welcome Create Server Wizard window appears. Click Next.

2 Select ECDA Option for Oracle. 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 a valid Admin Account name. Click Next.

5 Enter the Oracle connection string as previously defined in the *tnsnames.ora* file, and the path (including the file name) to the previously defined temporary location of the *tnsnames.ora* file. Click Next.

6 Verify the DirectConnect server information. If correct, select Create Server otherwise, click Back to return to the previous window and provide the correct information.

7 Verify if the DirectConnect configuration is successful. If yes, click Next otherwise, click Back to return to the previous window, provide the correct information, and recreate the server.

8 In the Start DirectConnect Server window, click Start *ecda_servername* to start the newly configured DirectConnect server, where *ecda_servername* is the server name you indicated for the new DirectConnect server.

9 Verify that the DirectConnect server startup is successful and click Finish.

❖ **To verify that you can connect to Oracle**

1 Open a command window in the *$SYBASE* directory of your ECDA installation.

2 Change to the *$SYBASE/DCO-15_0* directory.

3 Set the environment variables by sourcing the *$SYBASE/DCO_SYBASE.csh* file.

4 Use the isql utility to log in to Oracle through the DirectConnect server:

```
isql –Uvaliduser –Ppassword –SDCOServer
```

Here, *validuser* and *password* are any valid user and password, and *DCOServer* is the DirectConnect for Oracle server name.

5 Verify the connection to the replicate Oracle database:

```
select banner from v$version
go

BANNER
-----------------------------------------------
Oracle Database 10g Enterprise Edition Release
10.2.0.1.0 - Prod
PL/SQL Release 10.2.0.1.0 - Production
CORE 10.2.0.1.0 Production
TNS for Solaris Version 10.2.0.1.0 - Production
NLSRTL Version 10.2.0.1.0 - Production
(5 rows affected)
```

6 Exit the isql session with the quit command.

Creating a maintenance user for replication

This section explains how to create a maintenance user in Oracle.

Description of a maintenance user

The maintenance user is a valid Oracle user that the Replication Server uses to apply commands to the replicate Oracle database. Replication Server requires one maintenance user to be defined for each connection. Do not use the same name as the DCO Admin Account Name used in configuring the DirectConnect server.

❖ To create a maintenance user in Oracle

1. Using SQLPLUS, connect to the replicate database and enter:

   ```
   create user maintuser identified by password;
   ```

2. Grant dba and create session permissions to the maintenance user, who must have permission to issue SQL commands against any table to be replicated:

   ```
   grant dba to maintuser;
   grant create session to maintuser;
   ```

Creating objects and public synonyms

❖ To create objects and public synonyms

- Sign on to Oracle through ECDA Option for Oracle as the maintenance user (assuming that the maintenance user that you created has the resource role to create tables) and execute this script:

  ```
  isql -Umaintuser -Ppassword -SDCOServer
  -i $SYBASE/RAX-15_1/scripts/oracle
  /hds_oracle_new_setup_for_replicate.sql
  ```

  Here, *DCOServer* is the name of the DirectConnect server.

  Note You may receive a ORA-00942 message that a table or view does not exist. Ignore the message.

Verifying the ECDA installed objects

❖ To verify the ECDA installed objects

1. Open another command window in the $SYBASE directory of your ECDA installation.
2. Set the environment variables by sourcing the *$SYBASE/DCO-15_0/DCO_SYBASE.csh* file.
3. Use the isql utility to log in to Oracle through ECDA as *maintuser*.

   - To verify whether the rs_info table has been created successfully:

     ```
     isql -Umaintuser -Ppassword -SDCOServer
     select * from rs_info
     go
     ```

This 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:

```
isql -Umaintuser -Ppassword -SDCOServer
select * from rs_lastcommit
go
```

The following 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 are displayed, and you encounter an error message.

4 Exit the isql session with the quit command.

Configuring Replication Server

Configuring Replication Server includes the following sequence of tasks:

1 Configuring Replication Server for Oracle replication

2 Configuring Replication Server for replication to the target data server

3 Configuring Replication Server for replication from the primary database

Configuring Replication Server for Oracle replication

This section describes the configuration tasks for configuring Replication Server for Oracle.

❖ To apply heterogeneous datatype support (HDS) scripts to RSSD

Prepare the Replication Server RSSD for replication to an Oracle database. To apply HDS support scripts to RSSD:

1 Set the environment variables by sourcing *SYBASE.csh* file.

2 Change to the *$SYBASE/REP-15_0/scripts* directory:

3 Load the following scripts into the RSSD:

```
isql -U SAMPLE_RS_RSSD_prim -P SAMPLE_RS_RSSD_prim_ps –SSAMPLE_RS_ERSSD
–ihds_clt_ase_to_oracle.sql -DSAMPLE_RS_ERSSD
```

```
isql -U SAMPLE_RS_RSSD_prim -P SAMPLE_RS_RSSD_prim_ps –SSAMPLE_RS_ERSSD
–ihds_oracle_udds.sql -DSAMPLE_RS_ERSSD
```

```
isql -U SAMPLE_RS_RSSD_prim -P SAMPLE_RS_RSSD_prim_ps –SSAMPLE_RS_ERSSD
–ihds_oracle_funcstrings.sql -DSAMPLE_RS_ERSSD
```

Note The message “ASA -157 cannot convert to a timestamp” is displayed. You can ignore this message.

4 Go to the *$SYBASE/RAX-15_1/scripts/oracle* directory.

5 Load this script into the RSSD:

```
isql -U SAMPLE_RS_RSSD_prim
-P SAMPLE_RS_RSSD_prim_ps -SSAMPLE_RS_ERSSD
-ihds_oracle_new_udds.sql -DSAMPLE_RS_ERSSD
```

6 Shut down Replication Server:

```
isql –Usa –P –SSAMPLE_RS
shutdown
go
```

7 Navigate to the *$SYBASE/REP-15_0/samp_repserver* directory:

8 Run the *RUN_SAMPLE_RS* script, to start the SAMPLE_RS Replication Server.

Configuring Replication Server for replication to the target data server

This section describes the required configurations for Replication Server for replication to the target Oracle data server.

❖ To create a Replication Server connection to replicate

1 Navigate to the Replication Agent scripts *$SYBASE/RAX-15_1/scripts/oracle* directory.

2 Create the Oracle error class.

Execute these three scripts:

Note The *oracle_error_class_1_rs.sql* and *oracle_error_class_3_rs.sql* scripts are executed to the Replication Server. The *oracle_error_class_2_rssd.sql* script is executed in the RSSD.

```
isql -Usa -P -SSAMPLE_RS -i oracle_create_error_class_1_rs.sql

isql -U SAMPLE_RS_RSSD_prim -P SAMPLE_RS_RSSD_prim_ps -SSAMPLE_RS_ERSSD
-i oracle_create_error_class_2_rssd.sql -DSAMPLE_RS_ERSSD

isql -Usa -P -SSAMPLE_RS -i oracle_create_error_class_3_rs.sql
```

3 Make a copy of the *oracle_create_rs_standby_connection.sql* script:

```
cp oracle_create_rs_standby_connection.sql
my_oracle_create_rs_standby_connection.sql
```

4 Before executing the *my_oracle_create_rs_standby_connection.sql* script against your Replication Server, change all occurrences of value *{rds}.{rdb}* to the name of the connection that Replication Server uses to connect to DirectConnect server, where:

- *rds* is the DirectConnect server name.
- *rdb* is any valid identifier. Sybase recommends that you use the Oracle SID name.
- *rs_maint_user* and *rs_maint_user_pwd* are the maintenance user and password created in ECDA Option for Oracle, in step 2 of “Configuring ECDA Option for Oracle” on page 54.
- *maintuser* and *password* are the user name and password created in ECDA Option for Oracle.

For example:

```
create connection to DCOServer.oratest2
```

```
set error class oracle_error_class
set function string class rs_oracle_function_class
set username maintuser
set password “password”
set batch to “off”
go
```

Note In the above example, *password* is a RepServer reserved word and therefore, must be enclosed in double quotes.

5 Create the connection to the replicate database:

```
isql –Usa –P –SSAMPLE_RS -i my_oracle_create_rs_standby_connection.sql
```

6 To verify the Replication Server connection to the replicate database, log in to SAMPLE_RS and verify the Replication Server connection to the replicate database:

```
isql –Usa –P –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.”

7 Exit the isql session with the quit command.

Configuring Replication Server for replication from the primary database

This section describes the configuration steps required for replication from the primary database. It includes:

- Creating a Replication Server connection to the primary database
- Creating a database replication definition
- Creating the database replication subscription

❖ To create a Replication Server connection to the primary database

1 Go to the Replication Agent scripts, *$SYBASE/RAX-15_1/scripts/oracle* directory:

2 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
```

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

- *pds* is *rs_source_ds* (located in *$SYBASE/RAX-15_1/init/oracle.rs*).
- *pdb* is *rs_source_db* (located in *$SYBASE/RAX-15_1/init/oracle.rs*).

Note You must save the defined values to be used for the Replication Agent *oracle.rs* file later in the process.

For example, *NY.NYora92*.

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.

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

```
create connection to NY.NYora92
set error class rs_sqlserver_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 -P -SSAMPLE_RS -i my_oracle_create_rs_primary_connection.sql
```

A message is displayed that indicates the Replication Server connection to the primary database is created.

❖ To create the database replication definition

1 Navigate to the *$SYBASE/RAX-15_1/scripts/sybase* directory:

2 Make a copy of the *rs_create_test_db_repdef.sql* script:

```
cp rs_create_test_db_repdef.sql
```

```
my_rs_create_test_db_repdef.sql
```

3 Before executing the *my_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 *rs_source_ds* (located in *$SYBASE/RAX_15_1/init/oracle.rs*).
- *pdb* is *rs_source_db* (located in *$SYBASE/RAX_15_1/init/oracle.rs*).

For example, *NY.NYora92*

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 -P –SSAMPLE_RS -i my_rs_create_test_db_repdef.sql
```

A message is displayed that indicates the database replication definition is created.

❖ To create the database replication subscription

1 Make a copy of the *rs_create_test_db_sub.sql* script:

```
cp rs_create_test_db_sub.sql
my_rs_create_test_db_sub.sql
```

2 Before executing the *my_rs_create_test_db_sub.sql* script, change the "*{pds}.{pdb}*" and "*{rds}.{rdb}*'"to the appropriate connection name, where:

- *pds* is *rs_source_ds* (located in *$SYBASE/RAX_15_1/init/oracle.rs*).
- *pdb* is *rs_source_db* (located in *$SYBASE/RAX_15_1/init/oracle.rs*).
- *rds* is the DirectConnect server name
- *rdb* is any valid identifier. (Sybase recommends that you use the Oracle SID name.)

Here is an example of create database replication subscription:

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

```
go
```

3 Connect to Replication Server and execute:

```
isql -Usa -P –SSAMPLE_RS -i my_rs_create_test_db_sub.sql
```

A message is displayed that indicates the subscription is in the process of being created.

Configuring Replication Agent

Configuring Replication Agent involves the following sequence of tasks:

1. Configuring and verifying the primary Oracle database for replication
2. Creating the Replication Agent instance
3. Verify the Replication Agent instance
4. Initializing the Replication Agent instance

Configuring and verifying the primary Oracle database for replication

Before you install Replication Agent, configure the primary Oracle database for replication.

To ensure that the Oracle *redo* log data is not overwritten before it is read by the Replication Agent, you must complete the following tasks:

- Verify the current archiving setting of the redo logs.
- Verify the supplemental logging of primary key data.
- For Oracle 10g, verify that the flashback feature is disabled.
- Create an Oracle user and granting Oracle permissions.

❖ **To verify the current archiving setting of the redo logs**

Redo logs are used by Oracle to maintain a log infrastructure. The steps described below verify if the feature is turned on and, if not, indicate how to switch on the *redo* logging. This feature needs to be turned on only on the primary database.

1. Use SQLPLUS to connect to Oracle as a system administrator.
2. Run this command from SQLPLUS:

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

- If the archive log is on, the result should be:

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

3 To turn on log archiving, run these commands:

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

For more information about Oracle *redo* logs and archiving, see the Replication Agent *Primary Database Guide*.

❖ To verify the supplemental logging of primary key data

By default, Oracle does not log primary keys and unique indexes to its logging infrastructure. You must include the logging of these values for a successful replication of all table values. To verify whether this information is currently being recorded in the logging system:

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

- For Oracle 9, run this command from SQLPLUS:

  ```
  SELECT SUPPLEMENTAL_LOG_DATA_PK,
  SUPPLEMENTAL_LOG_DATA_UI FROM V$DATABASE;
  ```

  For Oracle 9, if logging of primary key and unique index values is enabled, the return values should be:

  ```
  SUP SUP

  --- ---

  YES YES
  ```

- For Oracle 10, run this command from SQLPLUS

  ```
  SELECT SUPPLEMENTAL_LOG_DATA_MIN,
  SUPPLEMENTAL_LOG_DATA_PK,
  SUPPLEMENTAL_LOG_DATA_UI
  FROM V$DATABASE;
  ```

  For Oracle 10, if logging of primary key and unique index values is enabled, the return values should be:

  ```
  SUP SUP SUP

  --- --- ---
  ```

```
            YES YES YES
```

2. If the result is different, turn on supplemental logging by executing these commands for Oracle 9 and Oracle 10:

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

❖ To verify that the Oracle 10g flashback feature is disabled

1. Use SQLPLUS to connect to Oracle as a system administrator with sysdba privileges.

2. View the current recycle bin configuration:

   ```
   select inst_id, value from v$parameter;
   ```

3. View the contents of the recycle bin:

   ```
   select * from dba_recyclebin;
   ```

4. Disable the recycle bin with the following commands:

   ```
   PURGE dba_recyclebin;
   ALTER SYSTEM SET recyclebin = OFF;
   ```

Note If you are using Oracle RAC, you must disable the recycle bin for each instance in the cluster.

❖ To create an Oracle user and grant permissions

- An Oracle database user must be created 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.

Use SQLPLUS to connect to the primary database as a system administrator and run the following commands to create an Oracle user named “RA_USER” 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 "CONNECT" TO "RA_USER";
GRANT "RESOURCE" TO "RA_USER";
GRANT "SELECT_CATALOG_ROLE" TO "RA_USER";
```

```
GRANT ALTER SESSION TO "RA_USER";
GRANT ALTER SYSTEM TO "RA_USER";
GRANT EXECUTE ON "SYS"."DBMS_FLASHBACK" TO
"RA_USER";
GRANT ALTER ANY PROCEDURE TO "RA_USER";
GRANT CREATE SESSION TO "RA_USER";
GRANT CREATE TABLE TO "RA_USER";
GRANT ALTER ANY TABLE TO “RA_USER”;
GRANT DROP ANY TABLE TO “RA_USER”;
GRANT CREATE PROCEDURE TO "RA_USER";
GRANT DROP ANY PROCEDURE TO “RA_USER”;
GRANT CREATE PUBLIC SYNONYM TO "RA_USER";
GRANT DROP PUBLIC SYNONYM TO "RA_USER";
GRANT SELECT ON SYS.OBJ$ TO "RA_USER";
GRANT SELECT ON SYS.LOB$ TO "RA_USER";
GRANT SELECT ON SYS.COLLECTION$ TO "RA_USER";
GRANT SELECT ON SYS.CON$ TO "RA_USER";
GRANT SELECT ON SYS.COLTYPE$ TO "RA_USER";
GRANT SELECT ON SYS.COL$ TO "RA_USER";
GRANT SELECT ON SYS.CDEF$ TO "RA_USER";
GRANT SELECT ON SYS.USER$ TO "RA_USER";
GRANT SELECT ON SYS.SEQ$ TO "RA_USER";
GRANT SELECT ON SYS.IND$ TO "RA_USER";
```

❖ **To verify the Oracle user roles created by the Replication Agent**

- As described in the previous section, the following Oracle commands can be issued to validate the required Oracle settings.

  Use SQLPLUS to run the following command as the new Oracle user:

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

  In addition, the user who starts the Replication Agent instance must 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.

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.This section describes how to prepare for the creation of an instance.

❖ To prepare the Replication Agent instance

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, usually found at

```
$ORACLE_HOME/jdbc/lib/ojdbc14.jar
```

Replication Agent may require a newer version of the JDBC driver, which you can obtain from at http://technet.oracle.com/software/tech/java/sqlj_jdbc/content.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 the following command to the .login script of a UNIX and Linux user:

```
setenv CLASSPATH /path_name/ojdbc14.jar:$CLASSPATH
```

For a Windows user, add the following to the CLASSPATH environment variable:

```
set CLASSPATH=path_name\jdbc\lib\ojdbc14.jar;
%CLASSPATH%
```

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 where an Oracle SQLPLUS session can be used to connect to the desired Oracle instance. The ORACLE_SID is from the environment variable, $ORACLE_SID.

Obtain host and port information from the file called *tnsnames.ora*, available at *$ORACLE_HOME/network/admin*.

Record the following:

- Host name of the Oracle database on which the TNS listener is operating.

- Port number the TNS listener is monitoring.
- ORACLE_SID value for the instance you wish to connect to.

4 (Optional) If your operating system has a Java 5.0.x-compatible JRE or JDK installed, you can use Oracle’s iSQL (the browser-based interface to SQL*PLUS) demo items to connect to Oracle using the JDBC driver and to verify the connection information.

To perform this validation, 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 under the Oracle installation directory:

- The *timezone.dat* binary file is the default file that contains the most commonly used time zones. This file is located in the *$ORACLE_HOME/oracore/zoneinfo* directory.
- The *timezlrg.dat* binary file contains a larger set of defined time zones. This file is located in the *$ORACLE_HOME/oracore/zoneinfo* directory.

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

Determine which *timezone* file is used by the primary Oracle instance and make a copy of that file available on the machine hosting the Replication Agent.

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 it 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*) can be found in the Replication Agent directory *$SYBASE/RAX-15_1/init*.

7 Create an instance resource file.

Copy the resource file template *$SYBASE/RAX-15_1/init/oracle.rs* to another file that contains the configuration information for a new instance, for example:

```
cp oracle.rs myra.rs
```

Using a text editor, 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 tnsname location, user, and password information for the Oracle, Replication Server, and RSSD connections.

The resource file is self-documenting, with each parameter described. For more information, refer to the Replication Agent *Administration Guide*, in Chapter 2, “Setup and Configuration,” the section titled, “Creating a Replication Agent instance using resource files.”

Table 2-4: 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)
pds_host_name	Machine (host) where Oracle is installed. **Note** To be used with pds_port_number, only if you have not set pds_tns_connection and pds_tns_filename.	jdoe_host1
pds_port_number	Port number for Oracle. **Note** To be used with pds_host_name, only if you have not set pds_tns_connection and pds_tns_filename.	1521
pds_database_name	$ORACLE_SID	NYora92

Parameter	Description	Example values
pds_tns_connection	Connection name found in the *tnsnames.ora* file which identifies the connection information for the primary database. **Note** To be used with pds_tns_filename, only if you have not set pds_host_name and pds_port_number.	ORA102.JDOE_HOST.COM
pds_tns_filename	File name identifying the Oracle *tnsnames.ora* file to be used to identify the connection information for the primary database. **Note** To be used with pds_tns_connection, only if you have not set pds_host_name and pds_port_number.	A valid Oracle *tnsnames.ora* file. For example: */opt/oracle/network/admin/ tnsnames.ora*
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.	jdoe_host1
rs_port_number	Port where Replication Server is installed.	11752
rs_username	Replication Server user with CONNECT SOURCE and CREATE OBJECT capabilities.	SAMPLE_RS_ra
rs_password	Password for rs_username.	SAMPLE_RS_ra_ps
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 Replication Server's configuration file, *$SYBASE/REP-15_0 /install /<server>.cfg*.	• Windows: cp850 • UNIX: iso_1
rs_source_ds	Valid name representing data server of Oracle primary database.	NY
rs_source_db	Valid name representing Oracle primary database.	NYora92
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

Parameter	Description	Example values
pdb_timezone_file	Path to the pdb_timezone_file directory that contains the archive redo log files.	*/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
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
pdb_archive_remove	Enables or disables the removal of archived Oracle redo log files from the path specified by pdb_archive_path.	USE_DEFAULT
asm_tns_connection	Identifies the Oracle ASM connection name found in the *tnsnames.ora* file.	A valid Oracle *tnsnames.ora* connection name **Note** To be used only if your Oracle environment uses ASM to manage the redo logs.
asm_tns_filename	Identifies the Oracle ASM *tnsnames.ora* filename where the connection information is located.	A valid Oracle *tnsnames.ora* file. **Note** To be used only if your Oracle environment uses ASM to manage the redo logs.
asm_username	Identifies the Oracle user name to be used when connecting to an ASM server.	User name used for ASM access **Note** To be used only if your Oracle environment uses ASM to manage the redo logs.
asm_password	Password for Oracle automatic Storage Management (ASM) access for the user specified in the asm_username.	Password used for the ASM user name **Note** To be used only if your Oracle environment uses ASM to manage the redo logs.

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, "To create a Replication Server connection to the primary database" on page 61.

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 Using the Replication Agent ra_admin utility, validate the settings in the resource file using the –vr parameter, for example:

```
$SYBASE/RAX-15_1/bin/ra_admin.sh -vr myra.rs
```

Here, *myra.rs* is the path name of the resource file. Validation results are returned as one of the following:

- `Response-file processing completed`
- `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 because of this process.

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, for example:

```
$SYBASE/RAX-15_1/bin/ra_admin.sh -r myra.rs
```

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`

For more information about the resource file or its use, refer to the Replication Agent *Administration Guide*, Chapter 2, “Setup and Configuration,” in the section titled, “Creating a Replication Agent instance using resource files.”

9 **Note** If in your resource file you set start_instance to yes and 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:

```
cd $SYBASE/RAX-15_1/myra
```

Execute the *RUN* file in the background, for example:

```
./RUN_myra &
```

Your Replication Agent for Oracle is now running.

Verify the Replication Agent instance

This section describes how to verify a Replication Agent instance, which includes the following sequence of tasks:

1 Update the Replication Server *interfaces* file with the Replication Agent location.

2 Verify the connection to the Replication Agent.

3 Verify the Replication Agent connection to Oracle.

❖ To verify the Replication Agent

1 Using dsedit, update the Replication Server *interfaces* file to include an entry for the Replication Agent location.

Note You can use any TDS client utility (isql, isqlApp, or SQLAdvantage) that you prefer.

2 Verify the connection to the Replication Agent:

a Open a command window in the *$SYBASE* directory of your Replication Server installation.

b Set the environment variables by sourcing the *SYBASE.csh* file.

c Use the isql utility to log in to Replication Agent:

```
isql –Usa –P –Smyra
```

Note The following verification steps are optional because they were done when you verified the resource file.

3 Verify the Replication Agent connection to Replication Server:

a Enter this command:

```
test_connection RS
go
```

b The following is displayed:

```
 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.

For more information, refer to the Replication Agent *Administration Guide*, Chapter 2, "Setup and Configuration," in the section titled, "Testing network connectivity."

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

a Enter this command:

```
test_connection PDS

go
```

This message appears:

```
Type Connection

 ---- ----------

 PDS  succeeded

(1 row affected)
```

b If the result indicates a failure:

- The server is not responding, or

- The connection properties; *tnsnames.ora* connection name 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.

For more information about connection failures, see the Replication Agent *Administration Guide* section titled, “Testing network connectivity.”

Initializing the Replication Agent instance

This section describes the commands you must issue to initialize a Replication Agent instance.

❖ **To initialize the Replication Agent instance**

1 **Note** If in your resource file you set initialize_instance to yes and no errors were received during the creation of the instance, skip this step.

The pdb_xlog 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 pdb_xlog 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:

```
 pdb_xlog init
 go
```

A message appears indicating that the procedure was successful.

2 Enter:

```
resume
go
```

If the Replication Agent successfully transfers to a replicating state, the following result is returned:

```
State Action
-------------------------------------
```

```
REPLICATING Ready to replicate data.
```

A returned state of ADMIN indicates that an error prevented the replicating state from being achieved. To determine the error, review the contents of the Replication Agent system log.

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.
```

A returned state of ADMIN indicates that an error prevented the replicating state from being achieved. To determine the error, review the contents of the Replication Agent system log.

4 Validate that both primary and replicate connections are active:

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

Note Be aware that:

- The DSI connection for the primary database connection is usually down 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 similar status for threads to the following:

```
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) DCOServer.oratest2
53   DSI        Awaiting Message 102 DCOServer.oratest2
17   SQM        Awaiting Message 102:0 DCOServer.oratest2DSI
20   DSI EXEC   Suspended        103(1) NY.NYora92
21   DSI        Suspended        103 NY.NYora92
24   DIST       Awaiting Wakeup  103 NY.NYora92
25   SQT        Awaiting Wakeup  103:1 DIST NY.NYora92
23   SQM        Awaiting Message 103:1 NY.NYora92
22   SQM        Awaiting Message 103:0 NY.NYora92
62   REP AGENT  Awaiting Command NY.NYora92
```

Testing replication

After you finish setting up the replication system, you can test replication.

❖ **To test replication**

1 Connect to the primary Oracle instance as a regular user, not 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 Create a test table in the replicate database:

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

5 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;
```

6 Connect to the Replication Agent and run the following command to mark the table:

```
pdb_setreptable T1,mark
```

7 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. See "Configuring Replication Server for replication from the primary database" on page 61.

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;
```

8 Use your preferred Oracle 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. Instead of deleting and re-creating a new Replication Agent instance, use this procedure to facilitate resetting the environment:

❖ To reset the primary database for replication

1 Protect your new environment from old log information by using the following command in the Replication Agent to archive all current *redo* log data:

```
pdb_xlog move_truncpt
```

2 To retain marking and log device information, reinitialize the Replication Agent using pdb_xlog init with the force option, which forces the Replication Agent repository to be refreshed instead of overwritten:

```
pdb_xlog init,force
```

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

3 Zero-out the locator stored in Replication Server using the rs_zeroltm command. For example:

```
isql -USAMPLE_RS_RSSD_prim -PSAMPLE_RS_RSSD_prim_ps
-SSAMPLE_RS_ERSSD -DSAMPLE_RS_ERSSD
```

```
rs_zeroltm NY, NYora92
go
```

The following is displayed:

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

CHAPTER 3

Setting Up Replication for IBM DB2 Universal Database

This chapter identifies the components that are required to implement a Sybase replication system for the IBM DB2 Universal Database on UNIX or Microsoft Windows. This chapter contains an overview of the tasks and instructions for installing and configuring a sample replication environment for IBM DB2 Universal Database.

Note This procedure is only for proof-of-concept (POC) or testing, not for production.

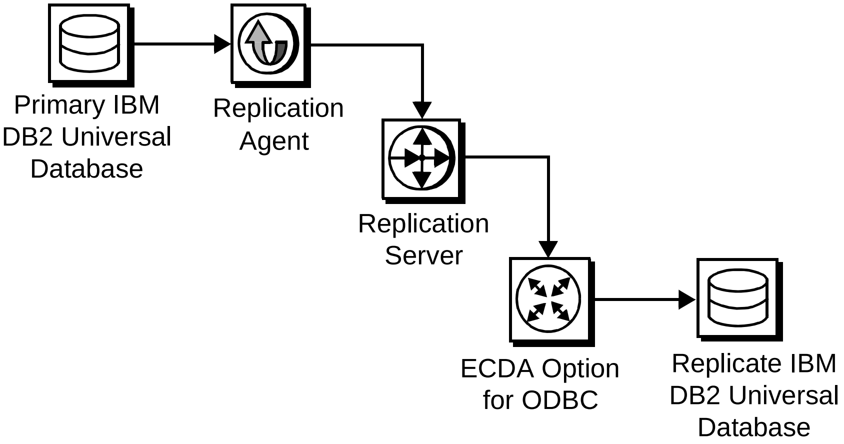
Sybase replication system

The components required to implement a replication system are:

- A primary IBM DB2 Universal Database
- A replicate IBM DB2 Universal Database
- A Replication Server
- The Replication Server Options components:
  - Replication Agent
  - ECDA Option for ODBC

Figure 3-1 illustrates a Sybase replication system with IBM DB2 Universal Database. It shows the flow of data between the databases and the Replication Agent, Replication Server, and ECDA database gateway.

Figure 3-1: Sybase replication system for IBM DB2 Universal Database


Primary IBM DB2 Universal Database
Replication Agent
Replication Server
ECDA Option for ODBC
Replicate IBM DB2 Universal Database

Prerequisites

Before you begin setting up the replication system, you should be familiar with the IBM DB2 Universal Database and have an understanding of Sybase replication. In addition, these conditions must be met:

- You have obtained the licenses for the Replication Server and the Replication Server Options.
- IBM DB2 Universal Database 8.2.2 or 9.1 is installed and configured, to serve as the source (primary) database and as the target (replicate) database.
- TCP/IP connectivity is available.
- Any OS patches required for Java 5.0 have been installed.

During this procedure:

- Because some directories, files, executable commands, and examples are shown only for Sun Solaris, you must adjust accordingly for Windows and other UNIX platforms.
- Sybase recommends that you do not use Replication Server reserved words for object names and connection names. A complete list of reserved words are documented in Chapter 2 of the Replication Server *Reference Manual*.

System requirements

The target computer on which you are installing the replication components must meet the following memory and disk space requirements:

For Windows

Table 3-1: Windows memory and disk requirements

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

For UNIX

Table 3-2: UNIX memory and disk requirements

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

For Linux

Table 3-3: Linux memory and disk requirements

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

Installing replication components

The following subsections describe how to install the replication components and how to verify their installation.

Note For the quick-start process, Sybase recommends installing all replication components on the same host where an IBM DB2 Universal Database server has already been installed and is running with both the primary and the replicate databases. This simplifies the quick-start process and avoids the step of separately installing IBM DB2 Universal Database clients for Replication Agent and for ECDA Option for ODBC.

Identifying the Sybase installation directory

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

```
/software/sybase
```

Accessing products from the SPDC Web site

If you are not using installation CDs, you must go to the Sybase product Download Center (SPCD) Web site to obtain replication software.

❖ **To download the replication software**

Note This procedure requires you repeat steps to download two products: Replication Server and Replication Server Options.

You must have a Login ID and Password to download software from the SPDC Web site.

1 Go to SPDC at https://sybase.subscribenet.com/control/sybs/login and enter your Login ID and Password.

The Product List page displays.

2 Select Replication Server.

The high-level Product Information page displays.

3 Select Replication Server again.

The detailed Product Information page displays.

4 Select the Replication Server software you need by primary database, version, and platform (return later to this detailed Product Information page to select Replication Server Options software later).

The Software Terms and Conditions page displays, listing countries and regions.

5 Either:

- Select I Agree to accept the terms without reading them, 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 displays. 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 go back to the high-level Product Information page.

6 The Product Download Page lists the software and documentation for the product. Select FTP Download 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 SyBooks download contains the rest of the documentation set.

7 Go back to the detailed Product Information page that you accessed in Step 3 and select the Replication Server Options software by primary database, version, and platform.

The Software Terms and Conditions page displays, listing countries and regions.

8 Navigate the Software Terms and Conditions pages as you did in Step 5.

9 At the Product Download Page, select FTP Download for the three components of Replication Server Options to download and save the compressed files to your local drive:

- Replication Agents
- ECDA Options
- DirectConnect Client

10 Uncompress and extract all the installation images to your local drive.

Installing Replication Agent

This section describes the steps for installing Replication Agent.

Note If Replication Agent is already installed, go to the next section called "Installing Replication Server" on page 89.

Installing Replication Agent

Installing the Replication Agent is a process that creates a directory structure containing the Replication Agent binaries, scripts, JRE, and associated files. There are no configuration steps during installation.

❖ To install Replication Agent software

1 Use *one* of the following methods to install the software.

- After following the download instructions from the SPDC Web site in the section "Accessing products from the SPDC Web site" on page 84, go to the location where you extracted the Replication Agent 15.1 for Sun Solaris installation image, or
- Insert the Replication Agent CD into your CD drive.

2 Launch the installation program by executing the setup routine for Solaris:

```
./setup
```

3 If you see an error message saying that there is not enough temporary disk space, add the following to your setup command:

```
./setup –is:tempdir someDirectory
```

Here, *someDirectory* is a directory with at least 100MB of disk space that the installation program can use.

Note If you are installing from a remote machine and do not have an X-windows client, execute ./setup –console, to run in console mode.

4 On the Welcome window, click Next.

5 Select a geographic location that displays the license agreement. Select I Agree and click Next.

6 Enter the directory where you want to install Replication Agent:

- If the directory does not exist, the installation program prompts you to create it. Click Yes.
- If the destination directory exists, you receive a warning message that you are installing into an existing directory. Click Yes.

7 Select Typical for the installation setup type and click Next.

8 The installation program displays a summary of what is to be installed. Click Next.

9 The installation program starts to install Replication Agent, displaying the progress status. If you see the following message;

```
There are newer files already installed and do you
want to replace them?
```

Click No to All.

10 When you receive a message that the installation was successful, click Next.

11 The SySAM License Server window opens and displays this prompt:

```
Will licenses be obtained from the License Server?
```

Respond based on how you set up your environment, use the license obtained from the Software Product Download Center (SPDC) Web site for Replication Server Options, and click Next.

12 The SySAM notification window prompts you to configure your server for e-mail notification. When configuration is enabled, you receive information about license management events that require attention. At the SySAM notification window, do one of the following:

- Select Yes and either accept the default values that are supplied, or enter values for the following:

    - SMTP server host name
    - SMTP server port number
    - Sender e-mail
    - Recipient e-mail
    - Message severity level of an event that triggers e-mail. Your choices are:
        - Informational
        - Warning
        - Error
  - If you choose not to have e-mail alerts or severity messages logged, select No.

  Click Next.

13. When you receive a message indicating that Replication Agent was successfully installed, click Finish.

For more information for the replication installation process, see the Replication Agent *Installation Guide*.

Verifying the installation

❖ To verify the installation

1. Change to the directory where you installed Replication Agent.
2. Set the environment variable by sourcing the *SYBASE.csh* file.
3. Change the directory to *$SYBASE/RAX-15_1/bin*.
4. Obtain the Replication Agent version string:

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

   If the Replication Agent installation completes successfully, you can see the Sybase copyright and the Replication Agent version string.

Installing Replication Server

This section describes the steps for a Replication Server installation and a procedure to verify the installation.

Note If Replication Server is already installed go to the next section called “Installing the ECDA Option for ODBC” on page 92.

Installation guidelines

While installing Replication Server, use the following guidelines:

- Allocate a disk partition of at least 20MB for each Replication Server you are installing. You can add more partitions later, if necessary. Check each partition to make sure it 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.

Installing Replication Server

❖ **To install Replication Server**

1. Use *one* of the following methods to download Replication Server software:

   - After following the download instructions from the SPDC Web site in the section “Accessing products from the SPDC Web site” on page 84, go to the location where you extracted the Replication Server 15.0 for Sun Solaris installation image, or

   - Insert the Replication Server CD in your CD-ROM drive.

2. Launch the installation program by executing the setup routine for Solaris:

   ```
   ./setup
   ```

3. If you see an error message saying that there is not enough temporary disk space, add the following to your setup command:

   ```
   ./setup –is:tempdir someDirectory
   ```

   Here, *someDirectory* is a directory with at least 100MB of disk space that the installation program can use.

4. On the Welcome window, click Next.

5 Select a geographic location that displays the license agreement. Select I agree and click Next.

6 Enter the directory where you want to install Replication Server:

- If the directory does not exist, the installation program prompts you to create it. Click Yes.
- If the destination directory exists, you receive a warning message that you are installing into an existing directory. Click Yes.

7 Select Typical for the installation setup type and click Next.

8 The installation program displays a summary of what is to be installed. Click Next.

9 The installation program starts to install Replication Server, displaying the progress status. You may see the following message:

```
There are newer files already installed and do you
want to replace them?
```

Click No to All.

10 When you receive a message indicating that Replication Server software has successfully installed, click Next.

Note This procedure uses the SAMPLE_RS Replication Server.

11 The SySAM License Server window opens and displays this prompt:

```
Will licenses be obtained from the License Server?
```

Respond based on how you set up your environment and use the license obtained from the Sybase Product Download Center (SPDC) Web site for Replication Server.

Click Next.

12 The SySAM notification window prompts you to 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 the following values from your worksheet:

- SMTP server host name
- SMTP server port number
- Sender e-mail

- Recipient e-mail
- Message severity level of an event that triggers e-mail. Your choices are:
  - Informational
  - Warning
  - Error

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

Click Next.

13. To start a sample Replication Server, select Yes and click Next.

14. This information is displayed and should be recorded:

```
This page contains detailed information regarding
the sample Replication Server. Please record this
information.

The sample Replication Server will be called
SAMPLE_RS and will run on port 11752. It will be
configured with a user of sa and no password.

The sample Replication Server will use an embedded
RSSD called SAMPLE_RS_ERSSD that runs on port 11751.
It will be configured with a user of
SAMPLE_RS_RSSD_prim and a password of
SAMPLE_RS_RSSD_prim_ps.

The installer has updated the appropriate interfaces
file or sql.ini file.

All files and logs associated with the sample
Replication Server will be located in the directory
$SYBASE/REP-15_0/samp_repserver.

The sample Replication Server will be configured
using the file $SYBASE/REP-15_0/samp_repserver
/SAMPLE_RS.res.
```

Click Next.

15. When the SAMPLE_RS is running and an “Installation was successful” message appears, click Finish.

Verifying the installation

❖ **To verify the installation**

1. Change to the directory where you installed Replication Server.
2. Set the environment variables by sourcing the *SYBASE.csh* file.
3. Use the isql utility to log in to Replication Server:

   ```
   isql -Usa -P -SSAMPLE_RS
   ```
4. Exit the isql session with the quit command.

Installing the ECDA Option for ODBC

This section describes how to install the ECDA Option for ODBC.

Note If ECDA Option for ODBC was already installed, go to the next section called “Configuring replication components” on page 94.

Installing ECDA Option for ODBC

When installing ECDA Option for ODBC, be sure to install ECDA on the same server as the replicate database. This eliminates a network hop and improves performance.

❖ **To install ECDA for ODBC in GUI mode**

Note If you want to install using the console mode or a response file, refer to the ECDA *Installation Guide* for Linux and UNIX.

1. Use *one* of the following methods to download the software:
   - After following the download instructions from the SPDC Web site as described in the section called “Accessing products from the SPDC Web site” on page 84, go to the location where you extracted the ECDA Option for ODBC 15.0 installation image.
   - Insert the CD labeled ECDA 15.0 Option for ODBC for Sun Solaris.
2. Launch InstallShield by executing the Sun Solaris setup routine:

   ```
   ./setup
   ```

3 If you see an error message saying that there is not enough temporary disk space, add the following to your setup command:

```
./setup –is:tempdir someDirectory
```

Here, *someDirectory* is a directory with at least 100MB of disk space that the installation program can use.

4 On the Welcome window, click Next.

5 Select a geographic location that displays the license agreement. Select I agree and click Next.

6 Enter the directory where you want to install ECDA:

- In the destination directory, enter the directory where the Replication Agent software is installed. Click Yes to install into an existing directory.
- If the destination directory exists, you receive a warning message that you are installing into an existing directory. Click Yes.

7 Select Custom setup, and click Next.

8 Deselect ECDA Option for Oracle, and click Next.

9 Verify the modules to install. Click Next.

You may receive a message:

```
There are newer files already installed and do you
want to replace them
```

Click No to All.

10 When you receive a message that the installation was successful, click Next.

11 The SySAM License Server window opens and displays this prompt:

```
Will licenses be obtained from the License Server?
```

Respond based on how you set up your environment and use the license obtained from the Software Product Download Center (SPDC) Web site for Replication Server Options.

Click Next.

12 The SySAM notification window prompts you to 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 the following (from your worksheet):

   - SMTP server host name
   - SMTP server port number
   - Sender e-mail
   - Recipient e-mail
   - Message severity level of an event that triggers e-mail. Your choices are:
     - Informational
     - Warning
     - Error

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

   Click Next.

13 When you receive a message indicating that ECDA was successfully installed, click Finish.

ECDA, including the DirectConnect server, is installed.

Configuring replication components

This section describes the tasks you must perform to configure each of the replication components in the replication system.

Setting IBM DB2 Universal Database environment variables

The IBM DB2 Universal Database installation provides two scripts, one of which needs to be sourced to set up the IBM DB2 Universal Database environment variables: *db2cshrc* for C shell and *db2profile* for Bourne or Korn shell. These scripts set library paths based on the bit size of the server or client that was installed.

For a 64-bit IBM DB2 Universal Database server or client, the paths point to the 64-bit libraries. However, except on HP Itanium, Replication Agent and ECDA require library paths to point to the 32-bit IBM DB2 Universal Database libraries. To correctly set the IBM DB2 Universal Database environment variables for Replication Agent and ECDA, you need to customize either of these script files. However, do not change the original scripts, because they may be changed when you apply a FixPak to the IBM DB2 Universal Database server or client.

❖ To set IBM DB2 Universal Database environment variables to use 32-bit libraries

1 Navigate to the IBM DB2 Universal Database server or IBM DB2 Universal Database client *sqllib* directory, which is located in the IBM DB2 Universal Database user home directory. For example:

```
cd ~dbsinst1/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 IBM DB2 Universal Database libraries. For example, on Solaris, add these lines:

- For C shell, in *mydb2cshrc*:

```
# force 32-bit libraries
setenv LD_LIBRARY_PATH $DB2DIR/lib32:$LD_LIBRARY_PATH
```

- For Bourne or Korn shell, in *mydb2profile*:

```
# force 32-bit libraries
LD_LIBRARY_PATH=$DB2DIR/lib32:$LD_LIBRARY_PATH
```

```
export LD_LIBRARY_PATH
```

4. Before starting either Replication Agent or ECDA, always source *your* file—rather than the original *db2cshrc* or *db2profile*—to set the IBM DB2 Universal Database environment variables.

Cataloging a node and a database in an IBM DB2 Universal Database client

To access their respective IBM DB2 Universal Databases, both the Replication Agent for UDB and ECDA Option for ODBC must use IBM DB2 Universal Database client libraries. The process of cataloging configures an IBM DB2 Universal Database client so that it knows how to connect to a particular database. This process also creates a database alias which Replication Agent or ECDA uses as the data source name.

By default, IBM DB2 Universal Database client libraries are installed when the IBM DB2 Universal Database server is installed. In addition, an IBM DB2 Universal Database client can be separately installed on a host other than the one on which the IBM DB2 Universal Database server is installed. In either case, cataloging must be done in the IBM DB2 Universal Database client that Replication Agent or ECDA is using. If Replication Agent and ECDA are installed on different hosts, they are using different IBM DB2 Universal Database clients, so the cataloging must be done in both of the IBM DB2 Universal Database clients.

If the IBM DB2 Universal Database client and the IBM DB2 Universal Database server are on the same host, and there is only one IBM DB2 Universal Database server running, cataloging may have been done when the IBM DB2 Universal Database instance was installed and the databases configured. If so, then no further cataloging needs to be done.

❖ **To catalog a node in an IBM DB2 Universal Database client**

1. Navigate to the IBM DB2 Universal Database server or IBM DB2 Universal Database client *sqllib* directory, which is located in the IBM DB2 Universal Database user home directory. For example:

   ```
   cd ~db2inst1/sqllib
   ```

2. Set the IBM DB2 Universal Database environment variables by sourcing the appropriate script:

   - For C shell:

     ```
     source db2cshrc
     ```

- For Bourne or Korn shell:

```
. db2profile
```

3 Start the IBM DB2 Universal Database command line processor (CLP):

```
db2
```

4 If necessary, log in to the operating system as the IBM DB2 Universal Database instance owner.

5 Determine if the desired IBM DB2 Universal Database node has already been cataloged. At the CLP prompt, enter:

```
list node directory
```

If a node has already been cataloged, depending on whether the node is local (first entry below) or remote (second entry below), you should get results similar to this:

```
Node 1 entry:

Node name                     = DB2NODE1
Comment                       =
Directory entry type          = LOCAL
Protocol                      = LOCAL
Instance name                 = DB2_01

Node 2 entry:

Node name                     = DB2NODE2
Comment                       =
Directory entry type          = LOCAL
Protocol                      = TCPIP
Hostname                      = 10.22.85.228
Service name                  = 50001
```

6 If your desired node is not listed, catalog it at the CLP prompt:

- To catalog a local node, issue this command:

```
CATALOG LOCAL NODE node_name INSTANCE db2_instance_name
```

where:

- *node_name* is the name of the node on the local host.
- *db2_instance_name* is the name of the associated DB2 instance.

- To catalog a remote node, issue this command:

```
CATALOG TCPIP NODE node_name REMOTE ip_address SERVER port_number
```

where:

- *node_name* is the name of the node on the remote host.
- *ip_address* is either the IP address or name of the host on which the remote node is running.
- *port_number* is the port on which the remote node is listening for client connections.

7 Verify that the node has been cataloged by issuing the list node directory command again.

❖ To catalog a database in an IBM DB2 Universal Database client

1 Determine if the database has already been cataloged. At the CLP command prompt, execute this command:

```
list database directory
```

Depending on whether the database is local (first entry below) or remote (second entry below), you should get results similar to this:

```
Database 1 entry:

Database alias                       = GRP_DB
Database name                        = GRP_DB
Local database directory             = /work/db2inst1
Database release level               = b.00
Comment                              = UTF-8(Unicode DB)
Directory entry type                 = Indirect
Catalog database partition number    = 0
Alternate server hostname            =
Alternate server port number         =


Database 2 entry:

Database alias                       = DB1V9_DS
Database name                        = DB1_V9
Node name                            = DB2NODE2
Database release level               = b.00
Comment                              =
Directory entry type                 = Remote
Catalog database partition number    = -1
Alternate server hostname            =
Alternate server port number         =
```

2 If your desired database is not listed or does not have a database alias, catalog it at the CLP prompt:

- To catalog a database using the default alias which is the same as the database name:

```
CATALOG DATABASE database_name AT NODE node_name
```

where:

  - *database_name* is the name of either the primary or replicate database.
  - *node_name* is the name of the node that contains the database.

- To catalog a database with a specific alias:

```
CATALOG DATABASE database_name AS my_alias AT NODE node_name
```

where:

  - *database_name* is the name of either the primary or the replicate database.
  - *my_alias* is your desired alias.
  - *node_name* is the name of the node that contains the database.

3 Verify that the database has been cataloged by issuing the list database directory command again.

4 To exit the DB2 CLP, enter the quit command.

Configuring ECDA Option for ODBC

Configuring ECDA Option for ODBC involves the following sequence of tasks:

1 Configuring and starting the DirectConnect server

2 Creating a maintenance user for replication

3 Creating objects and public synonyms

4 Verifying the ECDA installed objectss

Configuring and starting the DirectConnect server

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

- The DB2 client has been configured as described in “Cataloging a node and a database in an IBM DB2 Universal Database client” on page 96.

- The port to be used by the DirectConnect server is unused and available.

❖ Creating and configuring a DirectConnect server

1. Start the Create Server wizard with the following command:

   ```
   DCWizard.sh
   ```

2. When the Welcome Create Server Wizard window appears, click Next.
   The Create DirectConnect Server Options window appears.

3. Select the ECDA Option for ODBC option and click Next.
   The Server Name and Port Information window appears.

4. Enter the Server Name for the new DirectConnect server and the Port Number that you want the server to “listen” on. Click Next.
   The ECDA - ODBC Driver Manager Options window appears.

5. Select the unixODBC Driver Manager without a Driver Manager Option and click Next. The Driver Manager Information window appears.

6. Select the following from the browse files:

   - The path to the IBM DB2 Universal Database ODBC Driver file, for example:

     ```
     /software/IBM/db2_08_01/lib/libdb2.so
     ```

     Note Version 9.1 of the HP Itanium IBM DB2 Universal Database ODBC 64-bit driver has a new name (“libdb2o.so”).

   - The DB2 script file that must be sourced before the DirectConnect server is started. If you have customized the IBM DB2 Universal Database environment script as described in “Setting IBM DB2 Universal Database environment variables” on page 95, use the path to that script. Otherwise, use the path to the original IBM DB2 Universal Database script, either *db2cshrc* or *db2profile*.

     For example, to use your *customized* script for C shell, enter the following:

     ```
     source path/mydb2cshrc
     ```

     Here, *path* is the full path to the directory in which you saved your customized file.

     For example, to use the *original* IBM DB2 Universal Database script for Bourne or Korn shell, enter the following:

     ```
     . path/sqllib/db2profile
     ```

Here, *path* is the home directory of the owner of the replicate IBM DB2 Universal Database instance.

7 Click Next. The DirectConnect Service Name window appears.

8 Enter the ECDA access service name that you want to use for this server and click Next. The ECDA Server Summary and Build window appears.

9 Verify the ECDA Server information that appears:

- If correct, select Create Server. The Start DirectConnect server (Optional) window appears.
- If incorrect, click Back to change the information.

10 Using a text editor, open the *%SYBASE%\DC-15_0\servername\cfg\dcany.cfg* file, where *servername* is the name of the server you created in step 3 using the DCWizard.

11 In the file, locate the name of the IBM DB2 Universal Database access service that you created in step 4 above using the DCWizard. Under that, set the ConnectionSpec1 parameter to the name of the IBM DB2 Universal Database ODBC data source you created. For example:

```
[Service Library]
{Logging}
LogSvcLibStatistics=0
{Client Interaction}
SvclibDescription=Access Service Library for ODBC
[my_udb_access_service]
{ACS Required}
ConnectionSpec1=my_udb_datasource_name
```

12 Start the DirectConnect server:

a Navigate to the DirectConnect installation directory, for example:

```
cd /software/sybase/DC-15_0
```

b Execute the *DC_SYBASE.csh* script to set the DirectConnect environment variables.

c Navigate to the DirectConnect *bin* directory:

```
cd bin
```

d Start the DirectConnect server in the background, which also starts all the configured access services:

```
DCStart -Sservername &
```

Here, *servername* is the name of the DirectConnect server.

❖ **To verify that you can connect to IBM DB2 Universal 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 sourcing the *$SYBASE/DC_SYBASE.csh* file.

4. Use the isql utility to log in to the IBM DB2 Universal Database through the DirectConnect server:

   ```
   isql -Uvaliduser -Ppassword -SDCServer
   ```

   Here, *validuser* and *password* are any valid user and password, and *DCServer* is the DirectConnect server name.

5. Verify the connection to the replicate IBM DB2 Universal Database:

   ```
   select @@sqldbmsname
   go


   SQLDbmsName
   ---------------------
   DB2/SUN


   select @@sqldbmsver
   go


   SQLDbmsVer
   ---------------------
   08.02.0005
   ```

6. Exit the isql session with the quit command.

Creating a maintenance user for replication

This section explains how to create a maintenance user in IBM DB2 Universal Database.

The maintenance user is a valid IBM DB2 Universal Database user that the Replication Server uses to apply commands to the replicate IBM DB2 Universal Database. Replication Server requires one maintenance user to be defined for each connection.

A user must already exist on the operating system before it can be added to an IBM DB2 Universal 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 IBM DB2 Universal Database on Microsoft Windows. You may choose to install an IBM DB2 Universal Database client on a Windows machine and use it to administer the IBM DB2 Universal Databases.

❖ To create a maintenance user in IBM DB2 Universal Database

1. Select an existing operating system user to be added to the replicate database as a maintenance user.
2. Start the Control Center administration tool by choosing Start | Programs | IBM DB2 | Control Center.
3. Configure the Control Center administration tool to connect to the replicate IBM DB2 Universal Database according to the IBM documentation. Once the Control Center administration tool has been configured, the replicate database displays in the All Databases view.
4. In the Control Center administration tool display window, right-click on 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 POC environment, the user is created by an IBM DB2 Universal Database system administrator.

Creating objects and public synonyms

❖ To create objects and public synonyms

- Sign on to IBM DB2 Universal Database through ECDA Option for ODBC as the maintenance user (assuming that the maintenance user that you created has the resource role to create tables) and execute this script:

```
isql –Umaintuser –Ppassword –SDCOServer -i
$SYBASE/REP-15_0/scripts/hds_udb_setup_for_replicate.sql
```

Here, *DCOServer* is the name of the DirectConnect server.

Verifying the ECDA installed objects

❖ To verify the ECDA installed objects

1. Use the isql utility to log in to IBM DB2 Universal Database through ECDA as *maintuser*.

   - To verify whether the rs_info table has been created successfully:

     ```
     isql -Umaintuser -Ppassword -SDCOServer
     select * from rs_info
     go
     ```

     This 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:

     ```
     isql -Umaintuser -Ppassword -SDCOServer
     select * from rs_lastcommit
     go
     ```

     The following 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 are displayed, and you encounter an error message.

2. Exit the isql session with the quit command.

Configuring Replication Server

Configuring Replication Server involves the following sequence of tasks:

1. Configuring Replication Server for IBM DB2 Universal Database replication
2. Configuring Replication Server for replication to the target data server

3 Configuring Replication Server for replication from the primary database

Configuring Replication Server for IBM DB2 Universal Database replication

This section describes the configuration tasks for configuring Replication Server for UDB.

❖ To apply heterogeneous datatype support (HDS) scripts to RSSD

Prepare the Replication Server RSSD for replication to an IBM DB2 Universal Database. To apply HDS support scripts to RSSD:

1 Set the environment variables by sourcing *SYBASE.csh* file.

2 Change to the *$SYBASE/REP-15_0/scripts* directory:

3 Load these scripts into the RSSD:

```
isql -U SAMPLE_RS_RSSD_prim -P SAMPLE_RS_RSSD_prim_ps –SSAMPLE_RS_ERSSD
–ihds_clt_ase_to_udb.sql -DSAMPLE_RS_ERSSD

isql -U SAMPLE_RS_RSSD_prim -P SAMPLE_RS_RSSD_prim_ps –SSAMPLE_RS_ERSSD
–ihds_udb_udds.sql -DSAMPLE_RS_ERSSD

isql -U SAMPLE_RS_RSSD_prim -P SAMPLE_RS_RSSD_prim_ps –SSAMPLE_RS_ERSSD
–ihds_udb_funcstrings.sql -DSAMPLE_RS_ERSSD
```

4 Shut down Replication Server:

```
isql –Usa –P –SSAMPLE_RS
shutdown
go
```

5 Navigate to the *$SYBASE/REP-15_0/samp_repserver* directory:

6 Run the *RUN_SAMPLE_RS* script, to start the SAMPLE_RS Replication Server.

Configuring Replication Server for replication to the target data server

This section describes the required configurations for Replication Server for replication to the target IBM DB2 Universal Database data server.

❖ To create a Replication Server connection to replicate

1 Navigate to the Replication Server scripts *$SYBASE/REP-15_0/scripts* directory:

2 Make a copy of the *hds_udb_connection_sample* script:

```
cp hds_udb_connection_sample.sql
```

```
my_udb_rs_create_repl_connection.sql
```

3 Before executing the *my_udb_rs_create_repl_connection.sql* script against your Replication Server, change the name of the connection that Replication Server uses to connect to DirectConnect server, where:

- *udb_dcudb.udb_dcudb* is the DirectConnect server name followed by the replicate database name.
- *rs_maint_user* and *rs_maint_user_pwd* are the maintenance user and password created in ECDA Option for ODBC, in step 2 of "Configuring ECDA Option for ODBC" on page 99.

For example:

```
create connection to DCOServer.my_udb_db
set error class rs_sqlserver_error.class
set function string class rs_udb_function_class
set username maintuser
set password "password"
set batch to "off"
go
```

Note In the above 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 –P –SSAMPLE_RS -i my_udb_create_rs_repl_connection.sql
```

5 To verify the Replication Server connection to the replicate database, log in to SAMPLE_RS and verify the Replication Server connection to the replicate database:

```
isql –Usa –P –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

This section describes the configuration steps required for replication from the primary database. It includes:

- Creating a Replication Server connection to the primary database
- Creating a database replication definition
- Creating the database replication subscription

❖ To create a Replication Server connection to the primary database

1. Go to the Replication Agent scripts directory at *$SYBASE/RAX-15_1/scripts/sybase*:
2. Make a copy of the *rs_create_test_primary_connection.sql* script:

   ```
   cp rs_create_test_primary_connection.sql
   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 value *{rax}.{test}* to the name of the Replication Server connection between it and Replication Agent for UDB, where:

   - *rax* is rs_source_ds (located in *$SYBASE/RAX-15_1/init/ibmudb.rs*).
   - *test* is rs_source_db (located in *$SYBASE/RAX-15_1/init/ibmudb.rs*).

   Note You must save the defined values to be used for the Replication Agent *ibmudb.rs* file later in the process.

   For example, *NY.NYudb9.*

4. Change *sys* and *sys_pwd* to the user ID and password of the IBM DB2 Universal Database user who must have permission to apply DML operations against all user tables to be replicated, where:

   - *sys* is the user ID of the IBM DB2 Universal Database user (sys).
   - *sys_pwd* is the password of the IBM DB2 Universal Database user (change_on_install).

   Note This user must be a valid user in the IBM DB2 Universal Database.

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

   ```
   create connection to NY.NYudb9
   set error class rs_sqlserver_error_class
   set function string class rs_udb9_function_class
   set username sys
   set password change_on_install
   ```

```
with log transfer on, dsi_suspended
go
```

5 Execute this script in Replication Server:

```
isql –Usa –P –SSAMPLE_RS -i my_udb_rs_create_test_primary_connection.sql
```

A message is displayed that indicates the Replication Server connection to the primary database is created.

❖ To create the database replication definition

1 Navigate to the *$SYBASE/RAX-15_1/script/sybase* directory:

2 Make a copy of the *rs_create_test_db_repdef.sql* script:

```
cp rs_create_test_db_repdef.sql
my_udb_rs_create_test_db_repdef.sq
```

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 rs_source_ds (located in *$SYBASE/RAX_15_1/init/ibmudb.rs*).
- *pdb* is rs_source_db (located in *$SYBASE/RAX_15_1/init/ibmudb.rs*).

For example, *NY.NYudb9*.

This 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 -P –SSAMPLE_RS -i my_udb_rs_create_test_db_repdef.sql
```

A message is displayed that indicates the database replication definition is created.

❖ To create the database replication subscription

1 Make a copy of the *rs_create_test_db_sub.sql* script:

```
cp rs_create_test_db_sub.sql
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* is rs_source_ds (located in *$SYBASE/RAX_15_1/init/ibmudb.rs*).

- *pdb* is rs_source_db (located in *$SYBASE/RAX_15_1/init/ibmudb.rs*).
- *rds* is the DirectConnect server name
- *rdb* is any valid identifier. (Sybase recommends that you use the IBM DB2 Universal Database replicate database name.)

Here is an example of create database replication subscription:

```
create subscription NY_sub1
for database replication definition NY_repdef1
with primary at NY.NYudb9
with replicate at DCOServer.udb_db
without materialization
go
```

3 Connect to Replication Server and execute:

```
isql -Usa -P –SSAMPLE_RS -i my_udb_rs_create_test_db_sub.sql
```

A message is displayed that indicates the subscription is in the process of being created.

Configuring Replication Agent

Configuring Replication Agent involves the following sequence of tasks:

1. Configuring and verifying the primary IBM DB2 Universal Database for replication
2. Creating the Replication Agent instance
3. Verify the Replication Agent instance
4. Initializing the Replication Agent instance

Configuring and verifying the primary IBM DB2 Universal Database for replication

Before you install Replication Agent, configure the primary IBM DB2 Universal Database for replication.

❖ To verify the current archiving setting of the transaction log

Replication Agent requires that the primary IBM DB2 Universal Database transaction logging be archive logging, not circular logging.

1. Determine the LOGARCHMETH1 setting.

a Connect to the database by entering the following command at the CLP prompt:

```
CONNECT TO NYudb9 USER db2_user USING db2_user_ps
```

Here, *NYudb9* is the primary database, *db2_user* is the primary database user, and *db2_user_ps* is the password.

b Issue the following command to determine LOGARCHMETH1 setting:

```
GET DB CFG FOR dbalias
```

Here, *dbalias* is the cataloged alias of the primary database.

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, issue this command

```
UPDATE DATABASE CONFIGURATION USING logarchmeth1 LOGRETAIN
```

- To use a specific archive location, issue this command:

```
UPDATE DATABASE CONFIGURATION USING logarchmeth1 DISK:path
```

Here, *path* is the full path name of the directory where the archive logs are to be stored.

❖ To set the IBM DB2 Universal Database connectivity *autocommit* parameter

Replication Agent requires that the IBM DB2 Universal Database connectivity autocommit parameter be turned on (autocommit=1). The autocommit parameter is specified in the DB2 call-level interface (CLI) configuration file for the primary database. If the autocommit parameter is not turned on, a deadlock problem can occur.

- To set the autocommit parameter, edit the *db2cli.ini* file.
  It is located either in *$DB2DIR/cfg* (where $*DB2DIR* is the path to the IBM DB2 Universal Database client installation) or in *$HOME/sqllib* (where *$HOME* is the home directory of the IBM DB2 Universal Database instance owner of a IBM DB2 Universal Database server installation).

❖ To create an IBM DB2 Universal Database user and grant permissions

To connect to the primary IBM DB2 Universal Database and to read the IBM DB2 Universal Database transaction log, Replication Agent requires an IBM DB2 Universal Database user ID, and that user must have SYSADM or DBADM authority.

1. Create a new operating system user named *ra_user* using commands appropriate for your operating system.

   For example, the following command can be used to create a user named *ra_user* on the UNIX and Linux operating systems:

   ```
   useradd -gusers -Gmgmt -s/bin/shell -ppassword
   -d/home/ra_user -m ra_user
   ```

   Here, *password* is the password corresponding to the *ra_user* user name.

2. Start the Control Center administration tool by choosing Start | Programs | IBM DB2 | Control Center.

3. Configure the Control Center administration tool to connect to the primary IBM DB2 Universal Database according to the IBM documentation. Once the Control Center administration tool has been configured, the primary database displays in the All Databases view.

4. In the Control Center administration tool display window, right-click on 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 *ra_user* from the list of displayed operating system users, and click OK.

7. Click Grant All to add necessary authorities to *ra_user*.

8. Add a user temporary system-managed table space with a page size of at least 8KB.

   a. In the Control Center All Databases view, right-click the Table Spaces folder under your primary database, and select Create from the context menu.

   b. In the Name window, specify a name for the new table space, and select the Let DB2 Manage My Storage radio button. Click Next.

   c. In the Type window, select User Temporary, and click Next.

   d. In the Buffer Pool window, click Create.

   e. In the Create Buffer Pool window, specify a name for the buffer pool, specify 8KB as the page size in the drop-down menu, select Create Bufferpool Immediately, and click OK.

f Click Finish to create the table space. The new table space displays in the Table Spaces folder.

Note In a POC environment, the user is created by an IBM DB2 Universal Database system administrator.

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 IBM DB2 Universal Database that is to be replicated. This section describes how to prepare for the creation of an instance.

❖ To prepare the Replication Agent instance

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 it provides a way to record or retain the configuration information for a Replication Agent instance, allowing an instance to be removed and re-created Replication Agent instance, allowing an instance to be removed and re-created.

The resource file template for an IBM DB2 Universal Database instance is *$SYBASE/RAX-15_1/init/ibmudb.rs*.

2 Create an instance resource file.

Copy the resource file template *$SYBASE/RAX-15_1/init/ibmudb.rs* to another file that contains the configuration information for a new instance, for example:

```
cp ibmudb.rs myra.rs
```

Using a text editor, 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 IBM DB2 Universal Database, Replication Server, and RSSD connections.

The resource file is self-documenting, with each parameter described. For more information, refer to the Replication Agent *Administration Guide*, in Chapter 2, "Setup and Configuration," the section titled, "Creating a Replication Agent instance using resource files."

Note Table 3-4 lists parameters in the order they appear in the resource file, not alphabetically.

Table 3-4: 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)
pds_host_name	Machine (host) where IBM DB2 Universal Database is installed.	NY
pds_port_number	Port number for IBM DB2 Universal Database.	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 capabilities.	SAMPLE_RS_ra
rs_password	Password for rs_username.	SAMPLE_RS_ra_ps
rs_source_ds	Valid name representing data server of primary IBM DB2 Universal Database.	NY
rs_source_db	Valid name representing primary IBM DB2 Universal 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 Replication Server's configuration file, *$SYBASE/REP-15_0 /install/<server>.cfg*.	• Windows: cp850 • UNIX: iso_1
rssd_host_name	Machine where RSSD resides.	jdoe_host1

Parameter	Description	Example values
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 IBM DB2 Universal Database 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 IBM DB2 Universal Database transaction log files.	A valid directory path on the machine hosting Replication Agent that points to a location where IBM DB2 Universal Database 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 “To create a Replication Server connection to the primary database” on page 107.

3 Set the IBM DB2 Universal Database environment variables by sourcing either *db2chsrc* or *db2profile* as described in “Setting IBM DB2 Universal Database environment variables” on page 95.

4 Create and execute the new instance resource file:

a Using the Replication Agent ra_admin utility, validate the settings in the resource file using the –vr parameter, for example:

```
$SYBASE/RAX-15_1/bin/ra_admin.sh -vr myra.rs
```

Here, *myra.rs* is the path name of the resource file.

b Validation results are returned as one of the following:

- `Response-file processing completed`
- `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 because of this process.

c After the resource file has been validated, allow the ra_admin utility to create and configure the Replication Agent instance, using the –r parameter, for example:

```
$SYBASE/RAX-15_1/bin/ra_admin.sh -r myra.rs
```

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.

d Creation results are returned as one of the following:

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

For more information about the resource file or its use, refer to the Replication Agent *Administration Guide*, Chapter 2, “Setup and Configuration,” in the section titled, “Creating a Replication Agent instance using resource files.”

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:

```
cd $SYBASE/RAX-15_1/myra
```

Execute the *RUN* file in the background, for example:

```
./RUN_myra &
```

Your Replication Agent for UDB is now running.

Verify the Replication Agent instance

This section describes how to verify a Replication Agent instance, which involves the following sequence of tasks:

1 Update the Replication Server *interfaces* file with the Replication Agent location.

2 Verify the connection to the Replication Agent.

3 Verify the Replication Agent connection to IBM DB2 Universal Database.

❖ To verify the Replication Agent

1 Using dsedit, update the Replication Server *interfaces* file to include an entry for the Replication Agent location.

Note You can use any TDS client utility (isql, isqlApp, or SQLAdvantage) that you prefer.

2 Verify the connection to the Replication Agent:

a Open a command window in the *$SYBASE* directory of your Replication Agent installation.

b Set the environment variables by sourcing the *SYBASE.csh* file.

c Use the isql utility to log in to Replication Agent:

```
isql –Usa –P –Smyra
```

Note The following verification steps are optional because they were done when you verified the resource file.

3 Verify the Replication Agent connection to Replication Server:

a Enter this command:

```
test_connection RS
go
```

b The following is displayed:

```
 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.

For more information, refer to the Replication Agent *Administration Guide*, Chapter 2, “Setup and Configuration,” in the section titled, “Testing network connectivity.”

4 Verify the Replication Agent connection to the primary IBM DB2 Universal Database:

a Enter this command:

```
test_connection PDS
go
```

This message appears:

```
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 IBM DB2 Universal Database as the configured user to find which property is incorrectly defined.

For more information about connection failures, see the Replication Agent *Administration Guide* section titled, “Testing network connectivity.”

Initializing the Replication Agent instance

This section describes the commands you must issue to initialize a Replication Agent instance.

❖ To initialize the Replication Agent instance

1 **Note** If, in your resource file, you set initialize_instance to yes, skip this step and continue to step 2.

The pdb_xlog init command verifies that the primary IBM DB2 Universal Database is correctly configured to provide archived logging and that the IBM DB2 Universal Database user ID used by the Replication Agent has the necessary permissions. In addition, the pdb_xlog init command creates objects in the primary database to support replication.

To initialize the Replication Agent instance, run:

```
 pdb_xlog init
 go
```

A message appears indicating that the procedure was successful.

2 Enter this command:

```
resume
go
```

If the Replication Agent successfully transfers to a replicating state, the following result is returned:

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

A returned state of ADMIN indicates that an error prevented the replicating state from being achieved. To determine the error, review the contents of the Replication Agent system log.

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.
```

A returned state of ADMIN indicates that an error prevented the replicating state from being achieved. To determine the error, review the contents of the Replication Agent system log.

4 Validate that both primary and replicate connections are active:

```
isql -Usa -P -SSAMPLE_RS
admin who
```

```
go
```

Note Be aware that:

- The DSI connection for the primary database connection is usually down 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 has similar status for threads to the following:

```
admin who
go
```

The following is displayed:

```
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)
DCOServer.rdb

53 DSI Awaiting Message 102 DCOServer.rdb

17 SQM Awaiting Message 102:0 DCOServer.rdbDSI
EXEC Suspended 103(1) NY.NYudb9

DSI Suspended 103 NY.NYudb9

24 DIST Awaiting Wakeup 103 NY.NYudb9

25 SQT Awaiting Wakeup 103:1 DIST NY.NYudb9

23 SQM Awaiting Message 103:1 NY.NYudb9
```

```
22 SQM Awaiting Message 103:0 NY.NYudb9

62 REP AGENT Awaiting Command NY.NYudb9
```

Testing replication

After you finish setting up the replication system, you can test replication.

❖ To test replication

1 Connect to the primary IBM DB2 Universal Database instance as a regular user, not the maintenance user. Make sure the regular user also exists in the replicate database.

- To connect to the database with CLP, use the following command:

  ```
  CONNECT TO NYudb9 USER db2_user USING db2_user_ps
  ```

  Here, *NYudb9* is the primary database, *db2_user* is the primary database user, and *db2_user_ps* is the password.

2 Create a test table 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 Create a test table in the replicate database:

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

5 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
```

6 Connect to the Replication Agent and run the following command to mark the table:

```
pdb_setreptable T1, mark
```

7 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. See “Configuring Replication Server for replication from the primary database” on page 106.

8 Use your preferred IBM DB2 Universal Database 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 IBM DB2 Universal Database for replication

In a test environment, you may occasionally need to reset the replication environment. Instead of deleting and re-creating a new Replication Agent instance, use this procedure to facilitate resetting the environment:

❖ To reset the primary database for replication

1 Login into the Replication Agent and issue the suspend command:

```
suspend
go
```

2 Zero-out the locator stored in the Replication Agent by using the ra_locator command:

```
ra_locator zero
go
```

3 Zero-out the locator stored in Replication Server using the rs_zeroltm command, for example:

```
isql -USAMPLE_RS_RSSD_prim -PSAMPLE_RS_RSSD_prim_ps
-SSAMPLE_RS_ERSSD -DSAMPLE_RS_ERSSD


rs_zeroltm NY, NYudb9
go
```

This message is displayed:

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

4 Resume replication:

```
resume
go
```

Glossary

This glossary describes Replication Agent terms used in this book.

Adaptive Server

The brand name for Sybase relational database management system (RDBMS) software products.

- *Adaptive Server Enterprise* manages multiple, large relational databases for high-volume online transaction processing (OLTP) systems and client applications.
- *Adaptive Server IQ* manages multiple, large relational databases with special indexing algorithms to support high-speed, high-volume business intelligence, decision support, and reporting client applications.
- *Adaptive Server 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 database to a standby database in a single, atomic operation. No changes to primary data are allowed until the subscription data is captured at the primary database. See also **bulk materialization** and **nonatomic materialization**.

BCP utility

A bulk copy transfer utility that provides the ability to load multiple rows of data into a table in a target database. See also **bulk copy**.

bulk copy

An Open Client™ interface for the high-speed transfer of data between a database table and program variables. It provides an alternative to using SQL insert and select commands to transfer data.

bulk materialization

A materialization method whereby subscription data in a standby database is initialized outside of the replication system. You can use bulk materialization for subscriptions to table replication definitions or function replication definitions. 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**.

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 data to remote locations, and then keeping the replicated data synchronized with the primary data. Data replication is different from data distribution. Replicated data is stored copies of data at one or more remote sites throughout a system, and it is not necessarily distributed data. Contrast with **data distribution**. See also **disk replication** and **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**.

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 Replication Server to manage the database and distribute transactions to the database. Each database in a replication system can have only one database connection in Replication Server. See also **Replication Server** and **route**.

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

An abbreviation for *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 stand-alone data server system. Compare with **RDBMS**.

disaster recovery A method or process used to restore the critical business functions interrupted by a catastrophic event. A disaster recovery (or business continuity) plan defines the resources and procedures required for an organization to recover from a disaster, based on specified recovery objectives.

ERSSD An abbreviation for embedded *Replication Server System Database*, which manages replication system information for a Replication Server. See also **Replication Server**.

failback A procedure that restores the normal user and client access to a primary database, after a failover procedure switched access from the primary database to a standby database.See also **failover**.

failover A procedure that switches user and client access from a primary database to a standby 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 Replication Server object that represents a data server operation such as insert, delete, or begin transaction. Replication Server distributes operations to standby databases as functions. See also **function string**.

function string A string that Replication Server uses to map a function and its parameters to a data server API. Function strings allow Replication Server to support heterogeneous replication, in which the primary and standby 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 Replication Server to spool messages received from a Replication Agent. See also **outbound queue** and **stable queue**.

interfaces file A file containing information that Sybase Open Client and Open Server™ applications need to establish connections to other Open Client and Open Server applications. See also **Open Client** and **Open Server**.

isql An interactive SQL client application that can connect and communicate with any Sybase Open Server application, including Adaptive Server, Replication Agent, and Replication Server. See also **Open Client** and **Open Server**.

Java An object-oriented programming language developed by Sun Microsystems. A platform-independent, “write once, run anywhere” programming language.

Java VM The Java Virtual Machine. The Java VM (or JVM) is the part of the Java Runtime Environment (JRE) that is responsible for interpreting Java byte codes. See also **Java** and **JRE**.

JDBC An abbreviation for *Java Database Connectivity*. JDBC is the standard communication protocol for connectivity between Java clients and data servers. See also **data server** and **Java**.

JRE An abbreviation for *Java Runtime Environment*. The JRE consists of the Java Virtual Machine (Java VM or JVM), the Java Core Classes, and supporting files. The JRE must be installed on a machine to run Java applications, such as the Replication Agent. See also **Java VM**.

LAN An abbreviation for “local area network,” a computer network located on the user’s premises that covers 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 standby database. Specifically, latency is the time elapsed between committing an original transaction in the primary database and committing the replicated transaction in the standby 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 mirror (or standby) device.

See also **disk replication** and **transaction replication**.

LOB An abbreviation for *large object*, a type of data element that is associated with a column that contains extremely large quantities of data.

Log Reader An internal component of the Replication Agent that interacts with the primary database and mirror log devices to capture transactions for replication. See also **Log Transfer Interface** and **Log Transfer Manager**.

Log Transfer Interface An internal component of the Replication Agent that interacts with Replication Server to forward transactions for distribution to a standby database. See also **Log Reader** and **Log Transfer Manager**.

Log Transfer Manager An internal component of the Replication Agent that interacts with the other Replication Agent internal components to control and coordinate Replication Agent operations. See also **Log Reader** and **Log Transfer Interface**.

Maintenance User

A special user login name in the standby database that Replication Server uses to apply replicated transactions to the database. See also **Replication Server**.

materialization

The process of copying the data from a primary database to a standby database, initializing the standby database so that the Replication Agent system can begin replicating transactions. See also **atomic materialization**, **bulk materialization**, and **non-atomic materialization**.

nonatomic materialization

A materialization method that copies subscription data without a lock on the primary database. Changes to primary data are allowed during data transfer, which may cause temporary inconsistencies between the primary and standby databases. Contrast with **atomic materialization**. See also **bulk materialization**.

ODBC

An abbreviation for *Open Database Connectivity*, an industry standard communication protocol for clients connecting to data servers. See also **JDBC**.

Open Client

A Sybase product that provides customer applications, third-party products, and other Sybase products with the interfaces needed to communicate with Open Server applications. See also **Open Server**.

Open Client application

An application that uses Sybase Open Client libraries to implement Open Client communication protocols. See also **Open Client** and **Open Server**.

Open Server

A Sybase product that provides the tools and interfaces required to create a custom server. See also **Open Client**.

Open Server application

A server application that uses Sybase Open Server libraries to implement Open Server communication protocols. See also **Open Client** and **Open Server**.

outbound queue

A stable queue managed by Replication Server to spool messages to a standby database. See also **inbound queue** and **stable queue**.

primary data

The version of a set of data that is the source used for replication. Primary data is stored and managed by the primary database. See also **Replication Agent**, **primary database**, and **Replication Server**.

primary database

The database that contains the data to be replicated to another database (the standby database) through a replication system. The primary database is the database that is the source of replicated data in a replication system. Sometimes called the *active database*. Contrast with **standby database**. See also **primary data**.

primary key

The column or columns whose data uniquely identify 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 **standby 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

In a replication system, a state in which all updates have been propagated to their destinations. Some Replication Agent and 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

An abbreviation for *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

An abbreviation for *Replication Command Language*, the command language used to manage Replication Server.

RDBMS

An abbreviation for *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. See also **SQL**.

replicated data

A set of data that is replicated from a primary database to a standby database by a replication system. See also **primary database**, **replication system**, and **standby database**.

replicated transaction

A primary transaction that is replicated from a primary database to a standby database by a transaction replication system. See also **primary database**, **primary transaction**, **standby database**, and **transaction replication**.

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 a Replication Server for distribution to a standby database. See also **primary database** and **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 Replication Server, includes information about the columns to be replicated and the location of the primary table or stored procedure. See also **Replication Server** and **subscription**.

Replication Server

The Sybase software product that provides the infrastructure for a robust transaction replication system. See also **Replication Agent**.

RSSD

An abbreviation for *Replication Server System Database*, which manages replication system information for a Replication Server. See also **Replication Server**.

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 **disk replication** and **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**.

SQL

An abbreviation for *Structured Query Language*, a non-procedural 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 Replication Server. Messages written into the stable queue remain there until they can be delivered to the appropriate process or standby database. Replication Server provides a stable queue for both incoming messages (the inbound queue) and outgoing messages (the outbound queue). See also **database connection**, **Replication Server**, and **route**.

standby data

The data managed by a standby database, which is the destination (or target) of a replication system. See also **data replication** and **standby database**.

standby database

A database that contains data replicated from another database (the primary database) through a replication system. The standby database is the database that receives replicated data in a replication system. Sometimes called the *replicate database*. Contrast with **primary database**. See also **standby data**.

standby site

The location or facility at which standby data servers and standby databases are deployed to support disaster recovery, and normal business operations during scheduled downtime at the primary site. Sometimes called the *alternate site* or *replicate site*. Contrast with **primary site**. See also **standby database**.

subscription

A request for Replication Server to maintain a replicated copy of a table, or a set of rows from a table, in a standby database at a specified location. See also **replication definition** and **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**.

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 **Replication Server**.

transaction replication

A data replication method that copies data-changing operations from a primary database transaction log to a standby database. See also **data replication** and **disk replication**.

transactional consistency

A condition in which all transactions in the primary database are applied in the standby database, in the same order that they were applied in the primary database.

WAN

An abbreviation for “wide area network,” a system of local-area networks (LANs) connected together with data communication lines. Contrast with **LAN**.

Index