SYBASE®

Quick Start Guide

Mirror Activator™

15.0

[Linux, Microsoft Windows, and UNIX]

DOCUMENT ID: DC00714-01-1500-02

LAST REVISED: November 2007

Copyright © 2004-2007 by Sybase, Inc. All rights reserved.

This publication pertains to Sybase software and to any subsequent release until otherwise indicated in new editions or technical notes. Information in this document is subject to change without notice. The software described herein is furnished under a license agreement, and it may be used or copied only in accordance with the terms of that agreement.

To order additional documents, U.S. and Canadian customers should call Customer Fulfillment at (800) 685-8225, fax (617) 229-9845.

Customers in other countries with a U.S. license agreement may contact Customer Fulfillment via the above fax number. All other international customers should contact their Sybase subsidiary or local distributor. Upgrades are provided only at regularly scheduled software release dates. No part of this publication may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without the prior written permission of Sybase, Inc.

Sybase trademarks can be viewed at the Sybase trademarks page at http://www.sybase.com/detail?id=1011207. Sybase and the marks listed are trademarks of Sybase, Inc. ® indicates registration in the United States of America.

Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Unicode and the Unicode Logo are registered trademarks of Unicode, Inc.

All other company and product names used herein may be trademarks or registered trademarks of their respective companies.

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

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

Contents

About This Book

Sybase® Mirror Activator™ version 15.0 extends the capabilities of Replication Server® to support the following primary data servers in a Sybase replication system:

- Adaptive Server® Enterprise (ASE)
- Oracle Database Server

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 Mirror Replication Agent™ *Reference Manual* to look up detailed information about Mirror Replication Agent commands and configuration parameters.

This book is organized as follows:

Chapter 1, “Setting Up Replication for ASE,” describes the Quick Start procedure to install and configure a sample replication environment for ASE.

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

Related documents

Mirror Replication Agent Refer to these documents to learn more about the Mirror Activator:

- Mirror Activator *Administration Guide* – for an overview of the Mirror Activator, information about configuring and administering Mirror Replication Agent instances, and information about configuring the other components in a Mirror Activator system

- Mirror Activator *Primary Database Guide* – for detailed, database-specific information about each database that is supported by the Mirror Replication Agent
- Mirror Replication Agent *Installation Guide* – for information about installing the Mirror Replication Agent software
- Mirror Replication Agent *Reference Manual* – for information about asll Mirror Replication Agent commands and configuration parameters, including syntax, examples, and detailed command usage notes.
- Mirror Activator *Release Bulletin* – for last-minute information that was too late to be included in the documents

Note A more recent version of the Mirror Activator release bulletin may be available on the World Wide Web. To check for critical product or document information that was added after the release of the product CD, use the Sybase Technical Library Web site.

Replication Server Refer to these documents for more information about transaction replication systems and the Replication Server software:

- Replication Server *Design Guide* – for an introduction to basic transaction replication concepts and Sybase replication technology
- Replication Server *Administration Guide* – for guidelines for creating and managing a replication system
- Replication Server *Installation Guide* – for installation and upgrade procedures for all Replication Server and related products
- Replication Server *Reference Manual* – for the syntax and detailed descriptions of Replication Server commands
- Replication Server *Heterogeneous Replication Guide* – for detailed information about configuring Replication Server and implementing a Sybase replication system with Oracle databases

Enterprise Connect Data Access (ECDA) Refer to these documents for more information about ECDA:

- Enterprise Connect Data Access Option for Oracle *Server Administration and Users Guide* – for information about the features and functionality of the ECDA Option for Oracle
- Enterprise Connect Data Access *Installation Guide* – for installation and upgrade procedures for all ECDA and related products

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

- The Mirror 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 time frame 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
  ```

  where *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/MA-15_0/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, Mirror 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 Mirror Replication Agent commands. However, if you need to use a mixed-case object name in a Mirror 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.

Mirror Activator 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.

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 Mirror Activator, 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 ASE

Follow the instructions in this chapter to install and configure a *sample* replication environment for Adaptive Server Enterprise (ASE).

The procedure described in this chapter does **not** describe the steps for creating a mirrored copy.

Warning! This procedure is for Proof of Concept (POC) or test—*not* for production.

Prerequisites

Before using this procedure, you should be familiar with ASE and have an understanding of Sybase replication. In addition, these conditions must be met:

- The ASE primary and replicate servers are installed and running.
- Mirror Replication Agent has direct access to the ASE *transaction* logs or a mirrored copy of the *transaction* logs.

During this procedure:

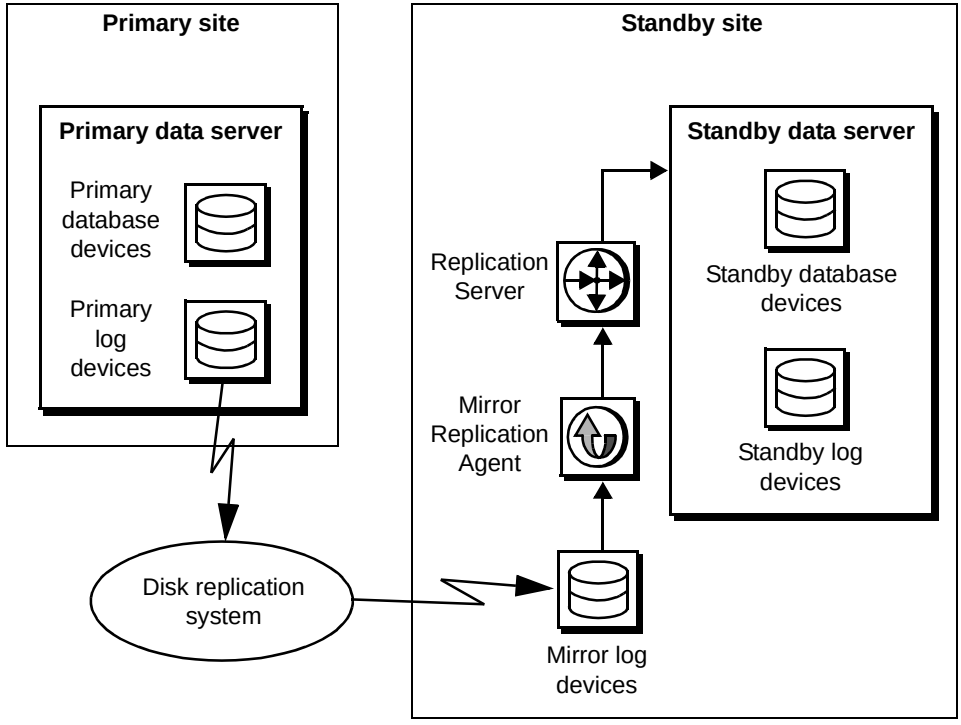
- Sybase recommends that you do not use Replication Server reserved words for object names and connection names. A complete list of reserved words are documented in Chapter 2 of the Replication Server *Reference Manual*.

- Because some directories, files, executable commands, and examples are shown only for Sun Solaris, you must adjust accordingly for Microsoft Windows and for other UNIX platforms.

System configuration

Figure 1-1 shows a typical Mirror Activator system configuration for ASE. Arrows illustrate the flow of mirrored log device data and replicated transactions during normal Mirror Activator system operation, that is, while replicating transactions from the primary database to the standby database.

Figure 1-1: Sybase replication system with ASE



Installing Mirror Activator components

This section describes how to install these components of Mirror Activator for ASE and how to verify their installation:

- Mirror Replication Agent
- Replication Server

Identify the Sybase installation directory

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

```
/software/sybase
```

Installing Mirror Replication Agent

This section describes the steps for a Mirror Replication Agent installation. It also has a procedure to verify the installation.

The Mirror Replication Agent installation process creates a directory structure containing the Mirror Replication Agent binaries, scripts, Java Runtime Environment (JRE), and associated files. There are no configuration steps during installation.

❖ **To install Mirror Replication Agent software**

1 Verify that the operating system (OS) patch levels are at the current level required to support Java 1.4.2. This is needed to support the JRE that is installed with Mirror Replication Agent.

   For more information about this requirement, see the Mirror Replication Agent *Installation Guide.*

2 Use **one** of the following methods to install the software:

   - From the Sybase Software Product Download Center (SPDC), navigate to the directory where the Mirror Activator software has been downloaded, uncompressed, and extracted. Download the Mirror Activator 15.0 for Sun Solaris, or
   - Insert the Mirror Activator CD into your CD-ROM drive.

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

```
./setupsolaris
```

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

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

where *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 ./setupsolaris –console to run in console mode.

5 On the Welcome window, click Next.

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

7 Enter the directory where you want to install Mirror Replication Agent:

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

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

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

10 The installation program starts to install Mirror Replication Agent, displaying the progress status. You might see this message:

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

Click No to All.

11 When you see a message that indicates that the installation was successful, click Next.

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

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

- If you answer Yes, you can leave the *license-mrx.lic* license file where it was installed (*$SYBASE/SYSAM-2_0/licenses*), or you can move the file to the *SYSAM-2_0/licenses* directory of your SYSAM license server. In either case, the Mirror Replication Agent will verify the necessary license.

- If you answer No, no further action is needed. The Mirror Replication Agent will use the installed license (*license-mrx.lic*) as it was installed in *$SYBASE/SYSAM-2_0/licenses*.

Click Next.

13 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 these values from your worksheet:

- SMTP server host name
- SMTP server port number
- E-mail return address
- Recipients of the notification
- Message severity level of an event that will trigger 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.

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

For more information about the installation process and the installation program, see the Mirror Replication Agent *Installation Guide.*

❖ **To verify the installation**

1 Change to the directory (*$SYBASE*) where you installed Mirror Replication Agent.

2 Set the environment variable by sourcing the *SYBASE.csh* file.

3 Change the directory to *$SYBASE/MA-15_0/bin*.

4 Obtain the Mirror Replication Agent version string:

```
ma.sh -v
```

You should see the Sybase copyright and the Mirror Replication Agent version string.

Installing Replication Server

This section describes the steps for a Replication Server installation. It also has a procedure to verify the installation.

Note If Replication Server is already installed, go to the next section called "Configuring Replication Server for replication to standby."

❖ **To install Replication Server**

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

- From the Sybase Software Product Download Center (SPDC), navigate to the directory where the Mirror Activator software has been downloaded, uncompressed, and extracted. Download the Mirror Activator 15.0 for Sun Solaris, or
- Insert the Mirror Activator CD into 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 this line to your setup command:

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

where *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 to create it.
- 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 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 to how you set up your environment and use the license obtained from the software product download site (SPDC) for Mirror Activator.

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 these values from your worksheet:

- SMTP server host name
- SMTP server port number
- Sender e-mail
- Recipient e-mails
- Message severity level of an event that will trigger 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 The SAMPLE_RS is now running and a “successful installation” message appears. Click Finish.

❖ **To verify the installation**

1 Change to the directory (*$SYBASE*) where you installed Replication Server.

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

3 Use isql to log in to Replication Server:

```
isql –Usa –P –SSAMPLE_RS
```

You should successfully log in to the Replication Server.

4 Exit isql.

Configuring Mirror Activator components for ASE

This section describes how to configure the components for Mirror Replication Agent for ASE:

- Configuring Replication Server
- Configuring Mirror Replication Agent

Configuring Replication Server

This section describes the required steps—shown in order—to configure Replication Server for ASE:

1 Configure the Replication Server for remote access

2 Configure Replication Server for replication to standby

3 Prepare the primary ASE for Replication Server

4 Configure Replication Server for replication from the primary database

For more information, see the Mirror Activator *Administration Guide* and the Mirror Replication Agent *Primary Database Guide*.

Configure the Replication Server for remote access

This section describes the required configurations to allow replication to occur when Replication Server is installed on a different (remote) machine than the primary or standby ASE servers.

The *interfaces* file entry for SAMPLE_RS and SAMPLE_RS_ERSSD uses "localhost" as the name of the machine where SAMPLE_RS and SAMPLE_RS_ERSSD reside. If the primary ASE database is not on the same machine as the SAMPLE_RS, the *interfaces* file entry must be changed to use either the machine name or the IP address of the machine.

Note This procedure is not required if the primary and standby ASE servers are on the same machine as Replication Server.

❖ **To configure the SAMPLE_RS for remote access**

1 Shut down Replication Server:

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

```
shutdown
go
```

A message displays indicating that Replication Server is shut down.

2 Edit the *interfaces* file.

3 Change the entry for the SAMPLE_RS and SAMPLE_RS_ERSSD from "localhost" to either a machine name or an IP address.

Note If the SAMPLE_RS is running, you receive a warning that the port is currently in use by a running server.

4 Add the primary ASE server and the standby ASE servers to the *interfaces* file, if they do not exist.

Note If the primary ASE server and standby ASE server are running, and you receive a warning that the port is currently in use by a running server., click OK. If the primary ASE server and standby ASE server are not running, start them.

5 Go to the *$SYBASE/$SYBASE_REP/samp_repserver* directory:

```
cd $SYBASE/$SYBASE_REP/samp_repserver/
```

6 Run the start-up script:

```
./RUN_SAMPLE_RS &
```

Configure Replication Server for replication to standby

This section describes the required configurations for Replication Server for replication to the standby ASE server.

❖ **To create a Replication Server connection to standby**

1 If needed, add the standby server to the *interfaces* file of the Replication Server and reboot the Replication Server.

2 Go to the *$SYBASE/MA-15_0/scripts/ase* directory.

3 Make a copy of the *ase_create_rs_standby_connection.rs* file:

```
cp ase_create_rs_standby_connection.rs
myma_ase_create_rs_standby_connection.rs
```

4 Using a text editor, change these values in the newly-created standby script and save the script:

- *$SYBASE* – enter the path name of the *$SYBASE* where the Replication Server resides, for example. */software/sybase*. (*$SYBASE* occurs three times in the script.)
- *$SYBASE_REP* – enter *REP-15_0*.
- *rds* – name of the data server where the standby database resides.
- *rdb* – name of the standby database.

Note

- If you are not using the SAMPLE_RS, change the SAMPLE_RS to the proper Replication Server name.
- The value entered for *rs.rs_ds_name* and *rs.rs_name* must exist in the *interfaces* file used by the Replication Server. The names are case sensitive.

5 Using the rs_init utility in Replication Server, create the connection from the Replication Server to the standby database:

```
$SYBASE/$SYBASE_REP/install/rs_init -r
myma_ase_create_rs_standby_connection.rs
```

The connection to the standby database is created.

Prepare the primary ASE for Replication Server

This section describes how to prepare the primary ASE database for Replication Server.

❖ **To prepare the primary ASE database for Replication Server**

- Add an entry for the Replication Server in the *interfaces* file of the primary ASE.

Note It is possible that a reboot will be required for the primary ASE. Sybase recommends that you reboot the ASE server to get a fresh copy of the *interfaces* file.

Configure Replication Server for replication from the primary database

This section describes these configuration steps required for replication from the primary database:

1 Configure Replication Server for replication from the primary database.

2 Grant create object permission to the Replication Agent user.

3 Create the database replication definition.

4 Create the database replication subscription.

❖ **To configure Replication Server for replication from the primary database**

1 If needed, add the primary ASE server to the *interfaces* file of the Replication Server and reboot the Replication Server.

2 Change the directory to *$SYBASE/MA-15_0/scripts*/ase.

3 Make a copy of *ase_create_rs_primary_connection.rs*:

```
cp ase_create_rs_primary_connection.rs
myma_ase_create_rs_primary_connection.rs
```

4 Using a text editor, change these values in the newly created script and save the script:

- *$SYBASE* – enter the path name of the *$SYBASE* where the Replication Server resides, for example, */software/sybase*. (*$SYBASE* occurs three times in the script).
- *$SYBASE_REP* – enter *REP-15_0*.
- *pds* – name of the ASE data server where the primary database resides.
- *pdb* – name of the primary database.

Note

- If you are not using the SAMPLE_RS, change the SAMPLE_RS to the proper Replication Server name.
- The value entered for *rs.rs_ds_name* and *rs.rs_name* must exist in the *interfaces* file used by the Replication Server, and the names are case sensitive.

5 Using the rs_init utility in Replication Server, create the connection from the Replication Server to the primary database:

```
$SYBASE/$SYBASE_REP/install/rs_init -r
myma_ase_create_rs_primary_connection.rs
```

The connection to the primary database is created.

6 Verify that the connections to the primary and standby servers have been made:

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

```
Spid Name       State               Info
 ---- ---------- ------------------- ---------------------------------
  23 DSI EXEC  Awaiting Command   101(1)SAMPLE_RS_ERSSD.SAMPLE_RS_ERSSD
  19 DSI      Awaiting Message    101 SAMPLE_RS_ERSSD.SAMPLE_RS_ERSSD
  18 SQM      Awaiting Message    101:0 SAMPLE_RS_ERSSD.SAMPLE_RS_ERSSD
  28 DSI EXEC   Awaiting Command    102(1) repit150i2.foo2
  27 DSI       Awaiting Message     102 repit150i2.foo2
  26 SQM       Awaiting Message     102:0 repit150i2.foo2
  38 DSI EXEC   Awaiting Command     103(1) repit150i.foo
  33 DSI       Awaiting Message     103 repit150i.foo
  35 DIST      Awaiting Wakeup      103 repit150i.foo
  36 SQT       Awaiting Wakeup     103:1  DIST repit150i.foo
  34 SQM       Awaiting Wakeup      103:1 repit150i.foo
  32 SQM       Awaiting Message     103:0 repit150i.foo
  37 REP AGENT  Awaiting Command     repit150i.foo
  20 dSUB       Sleeping
  15 dCM        Awaiting Message
  17 dAIO       Awaiting Message
  21 dREC       Sleeping            dREC
  39 USER       Active              sa
  14 dALARM     Awaiting Wakeup
   22 dSYSAM     Sleeping
```

In this example, *repit150i.foo* is the primary database, it has a Replication Agent thread, and *repit150i2.foo2* is the standby database. All threads for the primary and standby databases should have the “Awaiting Message” or “Awaiting Command” state.

❖ **To grant *create object* permission to the Replication Agent user**

1 Allow the default Replication Agent user to create replication definitions:

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

Permission is granted to user "SAMPLE_RS_ra."

Note Create object permissions must be granted to the value (located in *$SYBASE/MA-15_0/init/ase.rs*) that you use for the rs_username configuration parameter.

2 Exit isql.

❖ **To create the database replication definition**

1 Go to the *$SYBASE/MA-15_0/scripts/sybase* directory:

```
cd $SYBASE/MA-15_0/scripts/sybase
```

2 Make a copy of the *create_rs_db_repdef.sql* directory:

```
cp create_rs_db_repdef.sql
myma_create_rs_db_repdef.sql
```

3 Before executing the *myma_create_rs_db_repdef.sql* script, change the value of "{pds}.{pdb}" to the name of the connection string you defined for the primary database, where:

- *pds* is the name of the data server where the primary database resides.
- *pdb* is the name of the primary database.

For example:

```
repit150i.foo
```

Here is a database replication definition example:

```
create database replication definition repit150i_repdef1
with primary at repit150i.foo
replicate DDL
go
```

4 Execute the script in Replication Server:

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

A message appears indicating that the database replication definition is created.

❖ **To create the database replication subscription**

1 Make a copy of create_rs_db_sub.sql:

```
cp create_rs_db_sub.sql myma_create_rs_db_sub.sql
```

2 Using a text editor, change these values in the copied script and save the changes:
{*pds*} – name of the data server where the primary database resides
{*pdb*} – name of the primary database
{*rds*} – name of the data server where the standby database resides
{*rdb*} – name of the standby database
Here is a create database replication subscription example:

```
create subscription repit150i_sub1
   for database replication definition repit150i_repdef1
   with primary at repit150i.foo
   with replicate at repit150i2.foo2
   without materialization
```

3 Execute the script in Replication Server:

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

A message appears indicating that the subscription is being created.

Configuring Mirror Replication Agent

This section describes the required steps—shown in order—to configure Mirror Replication Agent for ASE:

1 Prepare and create the Mirror Replication Agent instance

2 Verify the Mirror Replication Agent instance installation

3 Resume the Mirror Replication Agent instance

4 Test replication

5 Reset the primary ASE database for replication

Prepare and create the Mirror Replication Agent instance

A single installation of the Mirror Replication Agent can support replication from multiple databases; however, one Mirror Replication Agent instance is needed for each ASE database to be replicated. This section describes how to prepare and create a Mirror Replication Agent for ASE instance:

❖ **To prepare for a Mirror Replication Agent instance**

1 Determine the default settings you want.

By default, Mirror Replication Agent is configured to mark every user table in the primary database for replication, and to create a replication definition for every table marked.

Although these settings may be reasonable for production environments, they may not be useful in a test or proof-of-concept environment, where only a subset of tables are being replicated. If automatic table marking or replication definition generation options are not wanted, change the settings of two Mirror Replication Agent configuration parameters: pdb_automark_tables and pdb_auto_create_repdefs.

For more information about the configuration parameters, refer to the Mirror Replication Agent *Reference Manual*.

2 Locate the sample *resource* file.

The majority of configuration values required to create and initialize a Mirror 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 Mirror Replication Agent for ASE instance, allowing an instance to be removed and re-created.

A sample resource file is located in the Mirror Replication Agent *$SYBASE/MA-15_0/init/ase.rs* directory.

❖ **To create a Mirror Replication Agent instance**

1 Go to the *$SYBASE/MA-15_0/init* directory.

2 Create an instance *resource* file. Copy the resource file template *$SYBASE/MA-15_0/init/ase.rs* file to another file that contains the configuration information for a new instance:

```
cp ase.rs myma.rs
```

3 Using a text editor, change the values assigned to the properties listed in Table 1-1:

Table 1-1: Resource file parameters

Parameter	Description	Example values
instance name	Any valid name.	myma
admin_port	Port number that Mirror Replication Agent will use.	20000
disable_rat	Disable Replication Agent thread.	yes

Parameter	Description	Example values
initialize_instance	Initialize the Mirror Replication Agent instance.	yes
pds_host_name	Machine (host) where ASE is installed.	repit
pds_port_number	Port number for ASE.	5600
pds_database_name	Name of database replicated from the primary data server.	foo
pds_username	A valid user name that has “sa” and replicate roles.	sa
pds_password	Password for pds_username.	sa_ps **Note** For null passwords, leave blank.
pds_sa_username	System admin name.	sa
pds_sa_password	System admin name password.	sa_ps **Note** For null passwords, leave blank.
ra_devicepath_1	Primary database log mirror location information.	*ra_devicepath_1= logA, /dev/pubs2/logA.dat*
rs_charset	Character set that Replication Server is using. **Note** The rs_charset value must be set to match the Replication Server character set.	iso_1
rs_host_name	Machine where Replication Server is installed.	repit
rs_port_number	Port where Replication Server is installed.	11752
rs_username	User ID that Mirror Replication Agent uses to access primary Replication Server; it must have CONNECT SOURCE and CREATE OBJECT capabilities.	SAMPLE_RS_ra
rs_password	Password for rs_username.	SAMPLE_RS_ra_ps

Parameter	Description	Example values
rs_source_ds	ASE server of the primary database.	repit150i
rs_source_db	Primary database.	foo
rssd_host_name	Machine where RSSD resides.	repit
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	Start the instance that was created.	yes

4 Validate and execute the new instance resource file:

```
$SYBASE/MA-15_0/bin/ma_admin.sh -vr myma.rs
```

where *myma.rs* is the path name of the *resource* file.

Validation results are returned as either:

- Response_file processing completed, or
- Response_file processing completed with errors.

  If any validation fails, the ma_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.

5 After the *resource* file has been validated, allow the *ma_admin* utility to create the MRA instance:

```
$SYBASE/MA-15_0/bin/ma_admin.sh -r myma.rs
```

A message appears, indicating that response file processing is completed.

Using the example settings in the *resource* file

Using the example settings specified in the *resource* file, the Mirror Replication Agent will:

- Stops disable, and preserve the secondary truncation point of the current ASE Mirror Replication Agent thread

- Mark the primary database with sp_reptostandby

Warning! If you have a large amount of text or image data in existing tables, the sp_reptostandby might take a long time to complete.

- Read all the tables in the primary database and prepare them for replication
- Initialize the Mirror Replication Agent

Verify the Mirror Replication Agent instance installation

This section describes the required steps to verify an Mirror Replication Agent instance:

❖ **To verify the Mirror Replication Agent installation**

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

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

2 Verify the connection to the Mirror 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 isql to log in to Mirror Replication Agent:

```
isql –Usa –P –Smyma
```

3 Verify the Mirror Replication Agent connection to Replication Server:

a Enter this command:

```
test_connection RS
go
```

b This output appears:

```
 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 additional information, refer to the Mirror Activator *Administration Guide*, Chapter 2, "Setup and Configuration," in the section titled, "Testing network connectivity."

4 Verify the Mirror Replication Agent connection to the primary ASE database:

a Enter this command:

```
test_connection PDS
go
```

This output appears:

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

b If the result indicates a failure:

- The server is not responding, or
- The connection properties host, port, pds_database_name, user, or password are incorrect.

Check the host and port configuration values, and manually log in to the primary ASE database as the configured user (pds_username in the *$SYBASE/MA-15_0/init/myma.rs*) to find which property is incorrectly defined.

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

Resume the Mirror Replication Agent instance

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

❖ **To resume the Mirror Replication Agent instance**

1 The resume command puts the Mirror Replication Agent in a *Replicating* state, reading the ASE *active* log and sending commands to Replication Server.

To place the Mirror Replication Agent in a *Replicating* state, enter the resume command:

```
resume

go
```

If the Mirror Replication Agent successfully transfers to a *Replicating* state, this result is returned:

```
State                                Action

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

REPLICATING (WAITING AT END OF LOG)  Ready to replicate data.
```

If the state returned is *Admin*, an error prevented the *Replicating* state from being achieved. To determine the error, review the contents of the Mirror Replication Agent instance log. Also, see the Mirror Activator *Administration Guide*, Chapter 2, "Troubleshooting Mirror Replication Agent," and the section titled, "Examining the Mirror Replication Agent when replication failure occurs."

2 The ra_status command returns the state of the Mirror Replication Agent. It is good practice to verify that the Mirror Replication Agent remains in *Replicating* state, even after the resume command executes successfully.

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

```
ra_status

go
```

If the Mirror Replication Agent is in *Replicating* state, this result is returned:

```
State                                 Action

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

 REPLICATING  (WAITING AT END OF LOG)  Ready to replicate data.
```

If the state returned is *ADMIN*, an error prevented the *Replicating* state from being achieved. To determine the error, review the contents of the Mirror Replication Agent instance log.

Also, see the Mirror Activator *Administration Guide*, Chapter 2, "Troubleshooting Mirror Replication Agent," and the section titled, "Examining the Mirror Replication Agent when replication failure occurs."

3 To validate that both primary and standby connections are active, issue the admin who command:

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

You should not proceed until admin who has a status for threads similar to this:

```
 Spid Name       State               Info
---- ---------- --------------- ---------------------------------
30 DSI EXEC   Awaiting Command    101(1) SAMPLE_RS_ERSSD.SAMPLE_RS_ERSSD
22 DSI       Awaiting Message    101 SAMPLE_RS_ERSSD.SAMPLE_RS_ERSSD
18 SQM       Awaiting Message    101:0 SAMPLE_RS_ERSSD.SAMPLE_RS_ERSSD
47 DSI EXEC    Awaiting Command    102(1) repit150i2.foo2
23 DSI         Awaiting Message     102 repit150i2.foo2
19 SQM         Awaiting Message     102:0 repit150i2.foo2
46 DSI EXEC   Awaiting Command     103(1) repit150i.foo
24 DSI        Awaiting Message     103 repit150i.foo
26 DIST        Awaiting Wakeup      103 repit150i.foo
29 SQT         Awaiting Wakeup      103:1  DIST repit150i.foo
21 SQM         Awaiting Message     103:1 repit150i.foo
20 SQM         Awaiting Message     103:0 repit150i.foo
53 REP AGENT  Awaiting Command     repit150i.f00
25 dSUB        Sleeping
15 dCM        Awaiting Message
17 dAIO        Awaiting Message
27 dREC        Sleeping               dREC
54 USER        Active                 sa
14 dALARM     Awaiting Wakeup
```

```
28 dSYSAM     Sleeping
```

Test replication

This section describes the steps required to test replication.

❖ **To test replication**

1 Connect to the primary ASE database as a user.

> **Warning!** Sybase recommends that you do *not* use the *maint user* because *maint user* filtering is enabled by default and transactions by this user will not be replicated.

2 Create a test table to replicate (unless it already exists):

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

After the elapsed time referenced by the Mirror Replication Agent setting in scan_sleep_max, the test table named "T1" should be replicated to the target ASE instance.

3 Grant permissions to any new or existing object to be replicated, so that the Replication Server maintenance user can update this table:

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

4 After the table is marked and the Mirror Replication Agent is in a *Replicating* state, insert test data into the test table and commit it:

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

5 Using your preferred ASE query tool, examine the standby site for results and compare the contents of your test table from both the primary and standby sites.

What to do if replication fails

If replication fails, refer to the Mirror Activator *Administration Guide*, in Chapter 2, "Troubleshooting Mirror Replication Agent," the section titled, "Examining the Mirror Replication Agent when replication failure occurs."

Reset the primary ASE database for replication

In a test environment, there may be times when the replication environment should be *reset*. Instead of deleting and re-creating a new Mirror Replication Agent instance, use this procedure to easily reset the environment.

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

1 Protect your new environment from old log information by using this command in the Mirror Replication Agent to move the transaction log truncation point to the end of the log:

```
pdb_init move_truncpt
```

2 To retain marking and log device information, re-initialize the Mirror Replication Agent using ra_init with the force option, which forces the Mirror Replication Agent repository to be refreshed instead of overwritten:

```
ra_init force
```

Note If you prefer to delete and replace all the information in the Mirror Replication Agent repository, issue the ra_deinit command followed by a normal ra_init command (without the force option):

For more information about the ra_init command, see the Mirror Replication Agent *Reference Manual*.

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 repit150i,foo

go
```

This message appears:

```
Locator has been reset to zero.

(return status = 0)
```

CHAPTER 2

Setting Up Replication for Oracle

Follow the instructions in this chapter to install and configure a *sample* replication environment for Oracle.

The procedure described in this chapter does **not** describe the steps for creating a mirrored copy.

Warning! This procedure is for Proof of Concept (POC) or test—*not* for production.

Prerequisites

Before using this procedure, you should be familiar with Oracle and have an understanding of Sybase replication. In addition, these conditions must be met:

- Mirror Replication Agent must have direct access to the Oracle **online** *redo* logs or a mirrored copy of those logs.
- Mirror Replication Agent must also have direct access to a copy of the Oracle **archived** *redo* logs or a mirrored copy of those logs.

During this procedure:

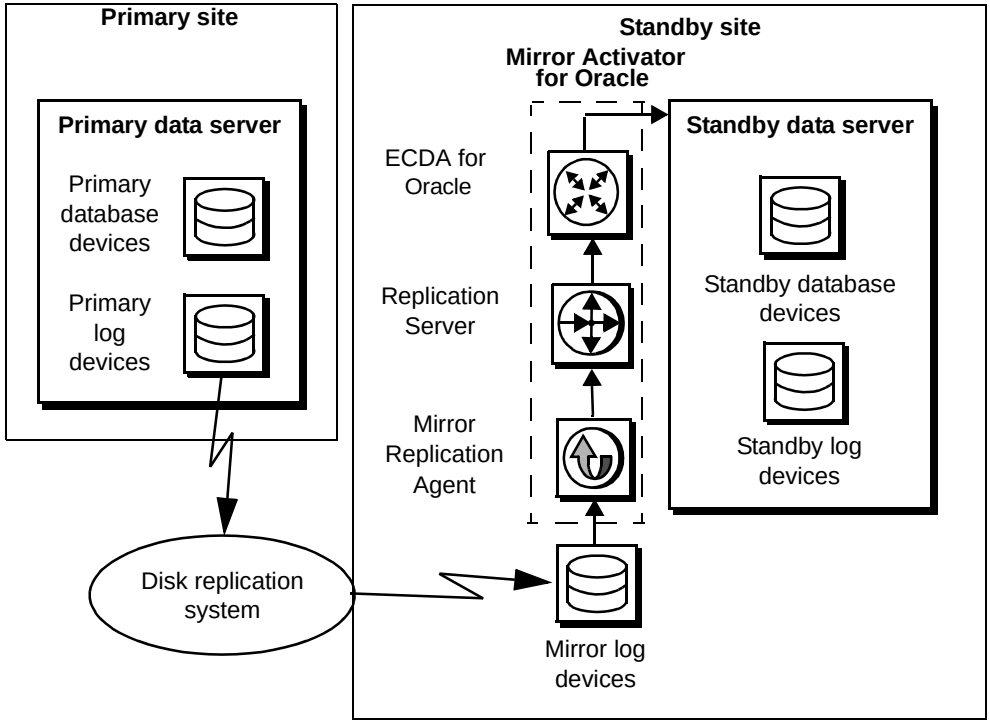
- Sybase recommends that you do not use Replication Server reserved words for object names and connection names. A complete list of reserved words are documented in Chapter 2 of the Replication Server *Reference Manual*.

- 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.

System configuration

Figure 2-1 illustrates a Sybase replication system with Oracle data servers. It shows the flow of data between the data servers, through the Mirror Replication Agent, Replication Server, and Enterprise Connect™ Data Access (ECDA) database gateway.

Figure 2-1: Sybase replication system with Oracle data servers



Installing Mirror Activator components

This section describes how to install these components of Mirror Activator for Oracle and how to verify the installation:

- Mirror Replication Agent
- Replication Server
- Enterprise Connect Data Access (ECDA) Option for Oracle, also referred to as Direct Connect for Oracle (DCO)

Identify the Sybase installation directory

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

```
/software/sybase
```

Installing Mirror Replication Agent

This section describes the steps for a Mirror Replication Agent installation. It also has a procedure to verify the installation.

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

❖ **To install Mirror Replication Agent software**

1 Verify that the operating system (OS) patch levels are at the current level required to support Java 1.4.2. This is needed to support the Java Runtime Environment (JRE) that is installed with Mirror Replication Agent.

   For more information about this requirement, see the Mirror Replication Agent *Installation Guide.*

2 Use **one** of the following methods to install the software:

   - From the Sybase Software Download Center Web site, navigate to the directory where the Mirror Activator software has been downloaded, uncompressed, and extracted. Download the Mirror Activator 15.0 for Sun Solaris, or
   - Insert the Mirror Activator CD into your CD drive.

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

```
./setupsolaris
```

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

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

where *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 ./setupsolaris –console to run in console mode.

5 On the Welcome window, click Next.

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

7 Enter the directory where you want to install Mirror 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.

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

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

10 The installation program starts to install Mirror Replication Agent, 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.

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

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

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

- If you answer Yes, you can leave both the *license-mrx.lic* and the *license-dco.lic* license files where they were installed in *$SYBASE/SYSAM-2_0/licenses*, or you can move the files to the *SYSAM-2_0/licenses* directory of your SYSAM license server. In either case, the Mirror Replication Agent and ECDA for Oracle (when it is subsequently installed) will verify the necessary licenses.
- If you answer No, no further action is needed. The Mirror Replication Agent and ECDA for Oracle will use the license files (*license-mrx.lic* and *license-dco.lic*) as they were installed in *$SYBASE/SYSAM-2_0/licenses*.

13 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 will trigger 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.

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

For more information, see the Mirror Replication Agent *Installation Guide*.

❖ **To verify the installation**

1 Change to the directory (*$SYBASE*) where you installed Mirror Replication Agent.

2 Set the environment variable by sourcing the *SYBASE.csh* file.

3 Change the directory to *$SYBASE/MA-15_0/bin*.

4 Obtain the Mirror Replication Agent version string:

```
ma.sh -v
```

You should see the Sybase copyright and the Mirror Replication Agent version string.

Installing Replication Server

This section describes the steps for a Replication Server installation. It also has 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."

❖ **To install Replication Server software**

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

- From the Sybase Software Product Download Center (SPDC) Web site, navigate to the directory where the Replication Server software has been downloaded, uncompressed, and extracted. Download the Replication Server 15.0.1 for Sun Solaris, 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 this to your setup command:

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

where *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 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 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 to how you set up your environment and use the license obtained from the Software Product Download Center (SPDC) for Mirror Activator.

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 these 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 will trigger 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.

❖ **To verify the installation**

1 Change to the directory (*$SYBASE*) where you installed Replication Server.

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

3 Use isql to log in to Replication Server:

```
isql –Usa –P –SSAMPLE_RS
```

4 Exit isql.

Installing the ECDA Option for Oracle

This section describes how to install the ECDA Option for Oracle and verify its installation.

Note If the ECDA Option for Oracle (also referred to as DCO) is already installed, go to the next section called "Configuring ECDA Option for Oracle."

Prepare to install ECDA Option for Oracle

This section describes how to install ECDA to replicate into Oracle.

❖ **To prepare the environment for ECDA Option for Oracle software**

1 Prepare the environment for installation:

- Review the Enterprise Connect Data Access *Installation Guide*, Chapter 6, for hardware, software, and operating system software requirements.
- Review sections 4 and 5 in the Enterprise Connect Data Access *Release Bulletin* for any recent updates to requirements for installation.

2 Identify an Oracle standby database. To do so, obtain the Oracle SID, which is usually the value $ORACLE_SID, and record the Oracle database name.

Note To verify that the Oracle version is certified to use with ECDA 15.0, see the certification report at http://certification.sybase.com/ucr/search.do.

3 Verify the complete path name to the *tnsnames.ora* file, for example:

```
$ORACLE_HOME/network/admin/tnsnames.ora
```

The *tnsnames.ora* file must be accessible to the machine where installation is taking place, because ECDA makes another copy of the *tnsnames.ora* file and places it under the *$SYBASE/DCO-15_0* directory structure during installation.

Install ECDA Option for Oracle

Note If the ECDA Option for Oracle (also referred to as DCO) was already installed and a directory was created:

- Set the environment variables by sourcing the *$SYBASE/SYBASE.csh* file and the *$SYBASE/DCO_SYBASE.csh* file.
- Execute the *$SYBASE/$SYBASE_DCO/install/DCConfig* to configure a new DCO and proceed to step 11.

❖ To install ECDA Option for Oracle in GUI mode

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

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

- From the Sybase Software Product Download Center (SPDC) Web site, navigate to the directory where the ECDA software has been downloaded, uncompressed, and extracted. Download the ECDA 15.0. Then, use the GNU utilities to unzip and un-tar the software download to a new directory. When un-tarred, the directory should contain these files: *media.inf*, *suite*, *archives*, *setup*.
- Insert the CD labeled ECDA 15.0 Options for Oracle for Sun Solaris.

2 Launch the installation program 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 this to your setup command:

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

where *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 Mirror 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 the Full setup and click Next.

8 Verify the modules to install. At a minimum, ECDA Option for Oracle, Connectivity, and Language Modules should be listed. Click Next.

You may receive a message:

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

Click No to All.

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

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

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

- If you answer Yes, you can leave both the *license-mrx.lic* and the *license-dco.lic* license files where they were installed in *$SYBASE/SYSAM-2_0/licenses*, or you can move the files to the *SYSAM-2/licenses* directory of your SySAM license server. In either case, the Mirror Replication Agent and ECDA Option for Oracle will verify the necessary licenses.
- If you answer No, no further action is needed. The Mirror Replication Agent and ECDA Option for Oracle will use the license files (*license-mrx.lic* and the *license-dco.lic*) as they were installed in the *SYBASE/SYSAM-2_0/licenses* directory.

11 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 these values from your worksheet (from the installation guide):

- SMTP server host name
- SMTP server port number
- E-mail return address
- Recipients of the notification
- Message severity level of an event that will trigger 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.

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

For more information, see the Enterprise Connect Data Access *Installation Guide*.

❖ **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 isql to log in to Oracle through the DirectConnect server:

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

where:

- *validuser* and *password* – any valid user and password.
- *DCOServer* is the DCO server name identified in the previous step.

5 Verify the connection to the standby 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 isql.

Configuring Mirror Activator components

This section describes the steps—shown in order—to configure Mirror Activator components for Oracle.

- Configuring ECDA Option for Oracle
- Configuring Replication Server
- Configuring Mirror Replication Agent

Configuring ECDA Option for Oracle

This section describes the steps—shown in order—to configure ECDA Option for Oracle:

1 Configure and start the DirectConnect server

2 Create a maintenance user and DDL user for replication

3 Create objects and public synonyms

4 Verify the ECDA installed objects

Configure and start the DirectConnect server

❖ **To configure and create a DirectConnect server**

1 Start the Create Server wizard.

- Windows – navigate to the *%SYBASE%\DCO-15_0\DCWizard* directory and execute the *DCWizard.bat* file.
- UNIX and Linux – 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 the ECDA Option for Oracle option. 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 ECDA 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 ECDA 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 ECDA server.

9 Verify if the ECDA server start-up is successful and click Finish.

Create a maintenance user and DDL user for replication

The maintenance user is a valid Oracle user that the Replication Server uses to apply commands to the standby Oracle database.

Note The maintenance user should not be the same as the DCO Admin Account Name used in configuring DCO.

❖ **To create a maintenance user in Oracle**

1 Using SQLPLUS (in the standby database), enter:

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

2 Grant permissions to the maintenance user, who must have the highest privilege of all users who have data or work that will be replicated:

- Grant dba to maintuser
- Grant create session to maintuser

❖ **To create a DDL user in Oracle**

1 Create a DDL (in the standby database) user in Oracle:

```
create user DDLuser identified by password;
```

Note The value for the *ddl_username* must not be the same as the value of the maintenance user defined in Replication Server for the standby connection.

2 Grant permissions to the DDL user, who must have privileges to perform DDL activities that can include create user, create, alter, drop procedures, indexes, tables and any DDL command issued on the correct database.

Note This user ID must have authority in the standby database to issue ALTER SESSION SET CURRENT_SCHEMA= *user* for any user who can create DDL in the primary database.

Create objects and public synonyms

❖ **To create objects and public synonyms**

- Create *rs_info table* and *rs_lastcommit* in the Oracle database as follows:
  - Sign on to Oracle through DCO 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/MA-15_0/scripts/oracle
/hds_oracle_new_setup_for_replicate.sql
```

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

Verify 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_SYBASE.csh* file.

3 Use isql to log in to Oracle through ECDA as maintuser:

```
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)
```

4 Exit isql.

Configuring Replication Server

This section describes the steps—shown in order—to configure Replication Server for Oracle:

1 Configure Replication Server for Oracle replication

2 Configure Replication Server for replication to standby

3 Configure Replication Server for replication from the primary database

For more information, see the Mirror Activator *Administration Guide* and the Mirror Replication Agent *Primary Database Guide*.

Configure Replication Server for Oracle replication

This section describes the required configurations for Replication Server for Oracle.

❖ **To apply Heterogeneous Datatype support (HDS) scripts to RSSD**

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

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

3 Change directories to the *$SYBASE/$SYBASE_REP/scripts* directory.

4 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" appears. You can ignore this message.

5 Go to the *$SYBASE/MA-15_0/scripts/oracle* directory.

6 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
```

7 Shut down Replication Server:

```
isql –Usa –P –SSAMPLE_RS

shutdown
go
```

A message appears indicating that Replication Server is shut down.

8 Go to the *$SYBASE/$SYBASE_REP/samp_repserver* directory.

9 Run the start-up script:

```
./RUN_SAMPLE_RS &
```

Configure Replication Server for replication to standby

This section describes how to configure Replication Server for replication to standby.

❖ **To create a Replication Server connection to standby**

1 Go to the *$SYBASE/MA-15_0/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_rs.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 directory*:

```
cp oracle_create_rs_standby_connection.sql
myma_oracle_create_rs_standby_connection.sql
```

4 Before executing the *myma_oracle_create_rs_standby_connection.sql* script against your Replication Server, change occurrences of value *{rds}.{rdb}* to the name of the connection that Replication Server will use to connect to DirectConnect for Oracle, where:

- *rds* is the DCO 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 DirectConnect for Oracle, in step 1 of configuring ECDA.
- *maintuser* and *password* are the user name and password created in DCO.

Here is an 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 previous example, "password" is a Replication Server reserved word, and it **must** be enclosed in double quotes.

5 Create the connection to the standby database:

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

The connection to the standby database is created.

6 To verify the Replication Server connection to the standby database, log in to SAMPLE_RS and verify the Replication Server connection to the standby 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 isql.

Configure Replication Server for replication from the primary database

This section describes how to configure for replication from the primary database:

- Create a Replication Server connection to the primary database.
- Create a database replication definition.
- Create the database replication subscription.
- Set up sequence replication support.
- Grant create object permission to Replication Agent user.
- Create a Replication Server user for Mirror Replication Agent to use.

❖ To create a Replication Server connection to the primary database

1 Verify that you are at the *$SYBASE/MA-15_0/scripts*/*oracle* directory.

2 Make a copy of the *oracle_create_rs_primary_connection.sql*:

```
cp oracle_create_rs_primary_connection.sql
```

```
myma_oracle_create_rs_primary_connection.sql
```

3 Before executing the *myma_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 to connect to Oracle, where:

- *pds* is *rs_source_ds* (located in *$SYBASE/MA-15_0/init/oracle.rs*).
- *pdb* is *rs_source_db* (located in *$SYBASE/MA-15_0/init /oracle.rs*).

For example, *NY.NYora92*.

4 Change *sys* and *sys_pwd* to the user ID and password of the Oracle user who will have permission to apply DML operations against all user tables that will 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 While not immediately used, this user must be a valid user in the Oracle database.

This command creates a Replication Server connection to the primary database example in step 3:

```
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 myma_oracle_create_rs_primary_connection.sql
```

A message appears indicating that the Replication Server connection to the primary database is created.

❖ **To create the database replication definition**

1 Go to the *$SYBASE/MA-15_0/scripts/sybase* directory:

```
cd $SYBASE/MA-15_0/scripts/sybase
```

2 Make a copy of the *create_rs_db_repdef.sql* file:

```
cp create_rs_db_repdef.sql
myma_create_rs_db_repdef.sq
```

3 Before executing the *myma_create_rs_db_repdef.sql* script, change the value of "{pds}.{pdb}" to the name of the connection string you defined for the primary database:

- *pds – rs_source_ds* (located in *$SYBASE/MA-15_0/init/oracle.rs*)
- *pdb – rs_source_db* (located in *$SYBASE/MA-15_0/init/oracle.rs*)

For example,

```
NY.NYora92
```

Here is an example of database replication definition:

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

4 Connect to the Replication Server and execute this command:

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

A message appears indicating that the database replication definition is created.

❖ **To create the database replication subscription**

1 Make a copy of the *create_rs_db_sub.sql* file:

```
cp create_rs_db_sub.sql myma_create_rs_db_sub.sql
```

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

- *pds* is *rs_source_ds* (located in *$SYBASE/MA-15_0/init/oracle.rs*).
- *pdb* is *rs_source_db* (located in *$SYBASE/MA-15_0/init/oracle.rs*).
- *rds* is the DCO server name (DCOServer).
- *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 this command:

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

A message appears indicating that the subscription is being created.

❖ **To set up sequence replication support**

1 Go to the *$SYBASE/MA-15_0/scripts/oracle* directory:

```
cd $SYBASE/MA-15_0/scripts/oracle
```

2 Install the rs_update_sequence stored procedure. Use SQLPLUS to run the oracle_create_replicate_sequence_proc.sql stored procedure at the standby site:

```
sqlplus maintuser/password @oracle_create_replicate_sequence_proc.sql
```

This is returned:

```
SQL*Plus: Release 10.2.0.1.0 - Production on Tue Mar
20 21:30:25 2007

Copyright (c) 1982, 2005, Oracle. All rights
reserved

Connected to:
Oracle Database 10g Enterprise Edition Release
10.2.0.1.0 - Production With the Partitioning, OLAP
and Data Mining options

Procedure created.

Grant succeeded.
```

Exit sqlplus.

For more information about sequence replication, see the Mirror Replication Agent *Primary Database Guide*.

3 To create a function replication definition for rs_update _sequence, copy the *oracle_create_rs_sequence_repdef.sql* script:

```
cp oracle_create_rs_sequence_repdef.sql
myma_oracle_create_rs_sequence_repdef.sql
```

4 Before executing the *myma_oracle_create_rs_sequence_repdef.sql* script, edit all occurrences of the value *{pds}.{pdb}* to the name of the Mirror Replication Agent connection used by your Mirror Replication Agent, where:

- *pds* is *rs_source_ds* (located in *$SYBASE/MA-15_0/init/oracle.rs*).
- *pdb* is *rs_source_db* (located in *$SYBASE/MA-15_0/init/oracle.rs*).

Here is the function replication definition for sequence replication after editing:

```
create function replication definition
REP_UPDATE_SEQUENCE
with primary at NY.NYora92
deliver as RS_UPDATE_SEQUENCE
(
@"SEQUENCENAME"  varchar(32768),
@"SEQUENCEVALUE" numeric,
@"INCREMENT"     numeric
)
send standby all parameters
go
```

5 Connect to the Replication Server and execute this command:

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

A message appears indicating that a functional sequence replication definition is created.

6 Use isql to log in to the Replication Server:

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

7 Verify that the database replication definition database subscription exists:

```
check subscription NY_sub1
for database replication definition NY_repdef1
with primary at NY.NYora92
with replicate at DCOServer.oratest2
go
```

A message appears indicating that a subscription is valid for the primary and replicate databases.

8 Replication Server comes with users who have the permissions necessary for Replication Agents. However, because Mirror Activator has the capability to create replication definitions, the Replication Agent user must also have CREATE OBJECT permission:

Note This user will be entered in the rs_username parameter when building the Mirror Replication Agent for Oracle instance. If you want a user other than the default replication agent user, make sure that the user is granted CONNECT SOURCE and CREATE OBJECT permission.

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

Permission is granted to user "SAMPLE_RS_ra."

Configuring Mirror Replication Agent

This section describes the steps—shown in order—to configure Mirror Replication Agent for Oracle:

1 Configure and verify the primary Oracle database for replication

2 Create the Mirror Replication Agent instance

3 Verify the Mirror Replication Agent instance installation

4 Initialize the Mirror Replication Agent instance

5 Test replication

6 Reset the primary Oracle database for replication

Configure and verify the primary Oracle database for replication

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

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

- Verify the current archiving setting of the *redo* logs
- Verify the supplemental logging of primary key data
- Create an Oracle user and grant Oracle permissions

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

1 Use SQLPLUS and 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
```

- To turn on log archiving:
  - Use SQLPLUS to connect to Oracle as a system administrator.
  - Run these commands from SQLPLUS:

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

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

❖ **To verify the supplemental logging of primary key data**

By default, Oracle does not log data from primary keys and unique indexes to the *redo* log. You must include the logging of these values for a successful replication of all table values.

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

2 Run this command from SQLPLUS:

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

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

```
SUP SUP

--- ---

YES YES
```

3 If the result is different, turn on supplemental logging by executing these commands:

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

ALTER DATABASE ADD SUPPLEMENTAL LOG DATA;
```

For more information about supplemental logging, see the Mirror Replication Agent *Primary Database Guide*.

❖ **To create an Oracle user and grant permissions**

An Oracle database user needs to be created for use by Mirror 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.

- From the following list of Oracle commands, create an Oracle user named "MA_USER" with the password "sybase," and grant permissions to the user:

```
CREATE USER "MA_USER" PROFILE "DEFAULT" IDENTIFIED
BY "sybase" DEFAULT TABLESPACE "USERS" ACCOUNT
UNLOCK;

GRANT "CONNECT" TO "MA_USER";

GRANT "RESOURCE" TO "MA_USER";

GRANT "SELECT_CATALOG_ROLE" TO "MA_USER";

GRANT ALTER SESSION TO "MA_USER";

GRANT ALTER SYSTEM TO "MA_USER";

GRANT EXECUTE ON "SYS"."DBMS_FLASHBACK" TO
"MA_USER";

GRANT ALTER ANY PROCEDURE TO "MA_USER";

GRANT CREATE SESSION TO "MA_USER";

GRANT CREATE TABLE TO "MA_USER";

GRANT CREATE PROCEDURE TO "MA_USER";

GRANT CREATE PUBLIC SYNONYM TO "MA_USER";

GRANT DROP PUBLIC SYNONYM TO "MA_USER";

GRANT SELECT ON SYS.OBJ$ TO "MA_USER";

GRANT SELECT ON SYS.LOB$ TO "MA_USER";
```

```
GRANT SELECT ON SYS.COLLECTION$ TO "MA_USER";

GRANT CREATE SEQUENCE TO "MA_USER";

GRANT SELECT ON SYS.CON$ TO "MA_USER";

GRANT SELECT ON SYS.COLTYPE$ TO "MA_USER";

GRANT SELECT ON SYS.COL$ TO "MA_USER";

GRANT SELECT ON SYS.CDEF$ TO "MA_USER";

GRANT SELECT ON SYS.USER$ TO "MA_USER";

GRANT SELECT ON SYS.SEQ$ TO "MA_USER";

GRANT SELECT ON SYS.IND$ TO "MA_USER";
```

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

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

- Using SQLPLUS, run this command as the new Oracle user:

```
sqlplus MA_USER/sybase

SQL*Plus: Release 10.2.0.1.0 - Production on Tue Mar
20 22:16:12 2007

Copyright (c) 1982, 2005, Oracle.  All rights
reserved.

Connected to:
Oracle Database 10g Enterprise Edition Release
10.2.0.1.0 - Production
With the Partitioning, OLAP and Data Mining options

select GRANTED_ROLE from USER_ROLE_PRIVS;

GRANTED_ROLE

--------------------
CONNECT
RESOURCE
SELECT_CATALOG_ROLE
```

In addition, the user who starts the Mirror Replication Agent for Oracle 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 Mirror Replication Agent is configured to remove old archive files, the user must have update authority to the directory and the *archive* log files.

Create the Mirror Replication Agent instance

A single installation of the Mirror Replication Agent can support replication from multiple databases; however, one Mirror Replication Agent instance is needed for each Oracle database to be replicated. This section describes how to prepare for an instance creation.

❖ **To prepare the Mirror Replication Agent instance**

1 Obtain the Oracle JDBC driver, which must be acquired from Oracle, and update the CLASSPATH 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*.

Mirror Replication Agent may require a newer version of the JDBC driver that was installed with Oracle. See the current version from Oracle at http://technet.oracle.com/software/tech/java/sqlj_jdbc/content.html.

Refer to the release bulletin or EBF letter of your Mirror Replication Agent to obtain the correct version of the JDBC driver.

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

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

Additional information about the JDBC driver requirements and setup can be found in the Mirror Replication Agent *Installation Guide*, Chapter 2, Setup and Configuration, in the section titled, “Setting up Mirror Replication Agent connectivity.”

3 Record the Oracle connectivity details for the primary Oracle 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, $ORACLE_SID. You can obtain host and port information from the file called *$ORACLE_HOME/network/admin/tnsnames.ora*.

Record the following:

- Host name of the Oracle database that the TNS listener is operating on (*$ORACLE_SID*)
- Port number the TNS listener is monitoring

- ORACLE_SID value for the instance you want to connect to

4 (*Optional*) If your operating system has a Java 1.4.x compatible JRE or JDK installed, you can use Oracle isql demo items to verify Java 1.4.x, and to connect to Oracle using the JDBC driver and the connection information.

To perform this validation, refer to the *Oracle JDBC Developer's Guide and Reference Guide*, in the section titled, "Verifying a JDBC Client Installation."

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

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

There are two *timezone* files under the Oracle installation directory:

- This binary file is the default file that contains the most commonly used time zones and is smaller:

  ```
  $ORACLE_HOME/oracore/zoneinfo/timezone.dat
  ```

- This binary file contains the larger set of defined time zones:

  ```
  $ORACLE_HOME/oracore/zoneinfo/timezlrg.dat
  ```

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 Mirror Replication Agent.

Note These files are platform dependent. You cannot use a *timezone* file from a little endian platform on a big endian platform, or vice versa.

---

6 Determine the default settings you want.

By default, Mirror Replication Agent is configured to mark every user table in the primary Oracle database for replication, and to create a replication definition for every table marked.

Although these settings may be reasonable for production environments, they may not be wanted in a test or proof of concept environment, where only a subset of tables are being replicated. If you do not want automatic table marking or replication definition generation options, change the settings of two Mirror Replication Agent configuration parameters: pdb_automark_tables and pdb_auto_create_repdefs.

For more information about these configuration parameters, refer to the Mirror Replication Agent *Reference Manual*.

7 Locate the sample *resource* file.

The majority of configuration values required to create and initialize a Mirror 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 Mirror Replication Agent instance, allowing an instance to be removed and re-created.

A sample resource file can be found in the Mirror Replication Agent directory *$SYBASE/MA-15_0/init/oracle.rs*.

8 Create an instance resource file.

Copy the *resource* file template *$SYBASE/MA-15_0/init/oracle.rs* to another file that contains the configuration information for a new instance, for example:

```
cp oracle.rs myma.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 Oracle, Replication Server, and RSSD connections.

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

Table 2-1: Resource file parameters

Parameter	Description	Example values
instance name	Any valid name	my_ma
admin_port	Port number that Mirror Replication Agent will use	9030 (if in use, select a different port number)
ddl_username	User name that can execute DDL commands	DDLuser
ddl_password	Password created for ddl_username.	password (created previously)
initialize_instance	Initialize the Mirror Replication Agent instance	yes
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 Mirror 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
pdb_include_archives	Enables or disables the use of Oracle archive log files	USE_DEFAULT
pdb_timezone_file	Path to the pdb_timezone-file directory that contains the archive *redo* log files	*/software/oracle/Ora10g/oracore /zoneinfo/timezone.dat*
pds_host_name	Machine (host) where Oracle is installed	crane
pds_port_number	Port number for Oracle	1521
pds_database_name	$ORACLE_SID	testoral
pds_username	Same as created previously in step 1, page 6	MA_USER
pds_password	Password for pds_username	sybase
pds_sa_username	System admin name	system
pds_sa_password	System admin name password	manager
ra_devicepath_1	Primary database log mirror location information	*ra_devicepath_1= logA, /dev/pubs2/logA.dat*, or leave commented if not using mirrored log

Parameter	Description	Example values
rs_charset	Character set that Replication Server is using **Note** The rs_charset value must be set to match Replication Server's character set.	iso_1
rs_host_name	Machine where Replication Server is installed	crane
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 Oracle primary database	NY
rs_source_db	Valid name representing Oracle primary database	NYora92
rssd_host_name	Machine where RSSD resides	crane
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	Start the instance that was created	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.

9 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 Mirror Replication Agent ma_admin utility, validate the settings in the *resource* file using the –vr parameter, for example:

```
$SYBASE/MA-15_0/bin/ma_admin.sh -vr myma.rs
```

where *myma.rs* is the path name of the *resource* file.

b Validation results are returned as either:

- Response_file processing completed, or
- Response_file processing completed with errors.

If any validation fails, the ma_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 ma_admin utility to create the Mirror Replication Agent instance, using the –r parameter, for example:

```
$SYBASE/MA-15_0/bin/ma_admin.sh -r myma.rs
```

d Validation results are returned as either:

- Response_file processing completed, or
- Response_file processing completed with errors.

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

10 Change to the *$SYBASE* directory and run the Mirror Replication Agent:

```
$SYBASE/MA-15_0/myma
```

Execute the *RUN* file, for example:

```
./RUN_myma &
```

Your Mirror Replication Agent for Oracle is now running.

Verify the Mirror Replication Agent instance installation

This section how to verify a Mirror Replication Agent instance:

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

2 Verify the connection to the Mirror Replication Agent.

3 Verify the Mirror Replication Agent connection to Oracle.

❖ **To verify the Mirror Replication Agent installation**

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

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

2 Verify the connection to the Mirror 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 isql to log in to Mirror Replication Agent:

```
isql –Usa –P –Smyma
```

3 Verify the Mirror 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 Mirror Replication Agent *Administration Guide*, Chapter 2, "Setup and Configuration," in the section titled, "Testing network connectivity."

4 Verify the Mirror 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; host, port, pds_database_name usually oracle_sid, user, or password are incorrect.

Check the host and port 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."

Initialize the Mirror Replication Agent instance

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

❖ **To initialize the Mirror Replication Agent instance**

1 The pdb_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 Mirror Replication Agent has the necessary permissions. It also creates objects in the database to support stored procedure replication.

The pdb_init command includes the move_truncpt keyword that executes an archiving operation in Oracle to archive all *redo* log files. This establishes the current position in the *redo* log as the end of the log, from a replication standpoint.

To verify correct configuration for archived and supplemental logging, issue the pdb_init command:

```
pdb_init move_truncpt

go
```

A message appears indicating that the procedure was successful.

For more information, see Chapter 2 in the Mirror Activator *Administration Guide*.

2 The ra_init command initializes the Replication Agent System Database (RASD) by reading schema information and *redo log* location information from the primary Oracle database. If this is a production setup, this step should coincide with creating the dump, copy, or data that is used to materialize the standby database.

To initialize the Mirror Replication Agent to read schema and *redo log* location information from the primary Oracle database, issue the ra_init command:

```
ra_init

go
```

A message appears indicting that the procedure was successful.

The ra_init command also causes pdb_automark_tables and pdb_auto_create_repdefs settings to take effect.

Note If this is a production setup, this step should coincide with creating the dump (copy) of data that is used to materialize the standby database.

---

3 The resume command puts the Mirror Replication Agent in an *Active* state, reading the Oracle *redo* log and sending commands to Mirror Replication Agent.

To place the Mirror Replication Agent in an *Active* state:

```
resume

go
```

If the Mirror Replication Agent successfully transfers to a *Replicating* state, this result is returned:

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

If the state returned is *Admin*, an error prevented the *Replicating* state from being achieved. To determine the error, review the contents of the Mirror Replication Agent system log. Also, see the Mirror Activator *Administration Guide*, Chapter 2, "Troubleshooting Mirror Replication Agent," in the section titled, "Examining the Mirror Replication Agent when replication failure occurs."

4 The ra_status command returns the state of the Mirror Replication Agent. It is good practice to verify that the Mirror Replication Agent remains in *Replicating* state, even after the resume command executes successfully.

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

```
ra_status
go
```

If the Mirror Replication Agent is in *Replicating* state, this result is returned:

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

If the state returned is *Admin*, an error prevented the *Replicating* state from being achieved. To determine the error, review the contents of the Mirror Replication Agent system log.

Also, see the Mirror Activator *Administration Guide*, Chapter 2, "Troubleshooting Mirror Replication Agent," in the section titled, "Examining the Mirror Replication Agent when replication failure occurs."

5 Verify that both primary and standby connections are active, using the admin who command:

```
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 Mirror Replication Agent connection, if established for the standby database connection, is usually down, because you are not replicating data *from* the standby database.

Do not proceed until admin who has similar status for threads to the following:

```
admin who
go
```

This 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.oratest2
53 DSI       Awaiting Message     102 DCOServer.oratest2
17 SQM       Awaiting Message     102:0 DCOServer.oratest2
DSI EXEC     Suspended            103(1) NY.NYora92
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
```

Test replication

This section describes the steps required to test replication.

❖ **To test replication**

1 Connect to the primary Oracle instance as a regular user.

2 Create a test table to replicate (unless it already exists):

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

This statement is replicated by the user defined in the Mirror Replication Agent configuration parameter ddl_user. After the elapsed time referenced by the Mirror Replication Agent setting in scan_sleep_max, the T1 table should be replicated to the target Oracle instance.

3 Grant permissions to any new or existing object to be replicated, so that the Replication Server maintenance user can update this table:

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

This statement is also replicated to the standby database by the ddl_user defined in the Mirror Replication Agent configuration.

4 After the table is marked and the Mirror 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 will not be replicated. The user ID used to insert transactions cannot be the same as the maintenance user defined in the primary connection. See the section titled, “Configure Replication Server for replication from the primary database” on page 43.

Mirror Replication Agent only applies committed transactions to a standby 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;
```

5 Using your preferred Oracle query tool, examine the standby site for results and compare the contents of your test table from both the primary and standby sites.

What to do if replication fails

If replication fails, refer to the Mirror Activator *Administration Guide*, in Chapter 2, “Troubleshooting Mirror Replication Agent,” the section titled, “Examining the Mirror Replication Agent when replication failure occurs.”

Reset the primary Oracle database for replication

In a test environment, there may be times when the replication environment should be *reset*. Instead of deleting and re-creating a new Mirror Replication Agent instance, use this procedure to facilitate resetting the environment.

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

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

```
pdb_init move_truncpt
```

Or, use the Oracle command:

```
SQL> alter system archive log current ;
```

2 To retain marking and log device information, re-initialize the Mirror Replication Agent using ra_init with the force option, which forces the Mirror Replication Agent repository to be refreshed instead of overwritten:

```
ra_init force
```

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

For more information about the ra_init command, see the Mirror Replication Agent *Reference Manual*.

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)
```

Glossary

This glossary describes Mirror Activator 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 Mirror 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, Mirror 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 Mirror 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 Mirror 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 Mirror 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 Mirror Replication Agent that interacts with the other Mirror Replication Agent internal components to control and coordinate Mirror 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 Mirror Activator 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 **Mirror 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 Mirror 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**.

Mirror Activator for Oracle — 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 **Mirror Activator for Oracle**.

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. Mirror Replication Agent reads the transaction log to identify and acquire the transactions to be replicated from the primary database. See also **Mirror 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

A

C

D

E

G

I

J

M

O

P

R

S

T

U

V