SYBASE®

An **SAP** Company

Configuration Guide

Replication Server® 15.7.1
SP100

UNIX

Contents

Contents

Conventions

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

Style conventions

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

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

Run:

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

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

Syntax conventions

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

Case-sensitivity

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

Terminology
Replication Agent™ is a generic term used to describe the Replication Agents for Adaptive Server® Enterprise, Oracle, IBM DB2 UDB, and Microsoft SQL Server. The specific names are:

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

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

Preparation for Installing and Configuring Replication Server

Prepare and organize the information you require for the replication system and Replication Server® you plan to install and configure.

Before you begin:

1. Plan your replication system.
2. Use the Replication Server Installation and Database Setup worksheets to collect and record the information you need to install and configure a Replication Server, and to add a database to your replication system.
3. Use the worksheets and the *Replication Server Installation Guide* to install Replication Server and the included sample Replication Server.

Plan the Replication System

Obtain the information you require to plan the replication system you intend to install and configure.

In your planning, you need to know:

- Where is the primary and replicate data?
- What Replication Servers will you need?
- Where is the RSSD for each Replication Server?
- Where is the initial disk partition for each Replication Server?
- Which Replication Server is the ID Server?
- Which databases require Replication Agents?
- Which databases require warm standby?

Primary and Replicate Data

Primary data is the set of data that is replicated from a source—the primary database, to another database—the replicate database, through the replication system. The set of data replicated from the primary database to the replicate database is the replicate data. The primary database is known to all of the Replication Servers with subscriptions for the data.

When planning your replication system, you need to:

- Determine which databases contain the data you want to replicate to other databases.

    Note: Although this document refers to Adaptive Server databases, you can use non-Adaptive Server databases with Replication Server.

• Determine the databases where Replication Server will maintain copies of the data.

See also
• *Install and Implement Non-ASE Support Features* on page 103

Required Replication Servers

When you design a replication system, you decide how many Replication Servers will be installed, which databases they will manage, and on which machine they will execute.

A Replication Server can manage one or more databases. One Replication Server is adequate for some replication systems. Other systems require a Replication Server at each geographic site to manage all of the databases at that site. Still others require multiple Replication Servers at each site to handle many databases or heavy transaction volumes.

When you upgrade Replication Servers, you can phase in the upgrades and install newer versions of Replication Servers initially only at certain sites. Consider the timing of such upgrades with care, and assess the need at each site for the new Replication Server features.

See also
• *Upgrade or Downgrade Replication Server* on page 55

Replication Server System Database Location

Replication system data is stored in a dedicated Adaptive Server Enterprise database called the Replication Server System Database (RSSD), or in an embedded Replication Server System Database (ERSSD) using SQL Anywhere®.

RSSD must be actively managed, while ERSSD is automatically created, configured, and maintained by Replication Server. If you choose to use RSSD, it is created when the Replication Server is installed.

Before you begin the installation, decide which Adaptive Server will store the RSSD for the Replication Server and the devices on which the database and its log will be stored. You can create the Adaptive Server database devices during installation, but identify their physical disk location before you install the Replication Servers.

If you use ERSSD, decide the directory paths for the ERSSD database file, transaction log, and backup. For better performance and recoverability, the database file, transaction log, and backup directory should be on different disk devices.

See also
• *Replication Server System Database Information* on page 20

Initial Disk Partition for Each Replication Server

Replication Server stores messages, such as replicated transactions, on disk in stable queues. Disk space for the queues is allocated from disk partitions you create. A disk partition can be a raw disk partition or an operating system file.

You must allocate a disk partition of at least 20MB for each Replication Server you install. You can add more partitions later if necessary. Check each partition to make sure that it is available and that the "sybase" user can write to it. To use an operating system file for a partition, create an empty file. Verify that there is enough space on the file system for Replication Server to extend the file to full partition size.

See *Replication Server Installation Guide > Before You Begin > Planning Your Installation > System Requirements*.

ID Server

The ID Server is a Replication Server that registers all Replication Servers and databases in the replication system.

In addition to the usual Replication Server tasks, the Replication Server acting as the ID Server assigns a unique ID number to every Replication Server and database in the replication system. The ID Server also maintains version information for the replication system. Otherwise, the ID Server is like any other Replication Server.

To allow a new Replication Server, or the Replication Server that manages the new database, to log in and retrieve an ID number, the ID Server must be running each time a:

- Replication Server is installed
- Route is created
- Database connection is created or dropped

Because of these requirements, the ID Server is the first Replication Server that you install and start when you install a replication system. If you have only one Replication Server, or if you are installing Replication Server for the first time, then that Replication Server is also the ID Server. If you are adding a Replication Server to an existing replication system, you must know the name of the Replication Server in the system that is the ID Server.

The ID Server must have a login name for Replication Servers to use when they connect to the ID Server. The login name is recorded in the configuration files of all Replication Servers in the replication system by the **rs_init** configuration program when you are setting up and managing the replication system.

Warning! The ID Server is critical to your replication environment, and is difficult to move once it has been installed. Once you have selected a name for the ID Server, you cannot change to a different Replication Server. Sybase does not support any procedures that change the name of the ID Server in the configuration files.

See also
*   *Replication Server Information* on page 14

Replication System Domain

Replication system domain refers to all replication system components that use the same ID Server.

Some organizations have multiple independent replication systems. Since the ID Server determines member Replication Servers and databases in a replication system, one replication system in an organization with multiple replication systems is also called an ID Server domain.

No special steps are required to set up multiple ID Server domains. Every Replication Server or database belongs to one replication system and has a unique ID number in that ID Server domain.

You can set up multiple replication system domains, with the following restrictions:

*   Replication Servers in different domains cannot exchange data. Each domain must be treated as a separate replication system with no cross-communication between them. You cannot create a route between Replication Servers in different domains.
*   A database can be managed by only one Replication Server in one domain. Any given database is in one, and only one, ID Server's domain. This means that you cannot create multiple connections to the same database from different domains.

Databases that Require Replication Agents

A Replication Agent retrieves transactions from a database log and passes them to the replication system through the Replication Server that manages the database.

You can configure multiple replication paths from Adaptive Server with an Adaptive Server Replication Agent (RepAgent) for each replication path. See *Replication Server Administraton Guide Volume 2 > Performance Tuning > Multi-Path Replication*. For non-Sybase databases there is only one Replication Agent per database. Because you must configure the Replication Agent when you set up a database for replication, you must know whether the database requires a Replication Agent before you add the database to the replication system. A user database requires a Replication Agent if:

*   The database holds primary data that is replicated to other databases managed by the same or different Replication Servers.
*   The Replication Server manages a database that is a source of replicated stored procedure execution.
*   You plan to replicate a stored procedure execution from a primary to replicate databases or from a replicate to a primary database. Replication Agent retrieves the procedure call information from the database transaction log. See the *Replication Server Administration Guide Volume 1*.

- The database is the active or standby database in a warm standby application. Replication Agent for a standby database is needed only when you switch to the standby database. See the *Replication Server Administration Guide Volume 1*.

An RSSD requires a Replication Agent if the Replication Server using that RSSD has a route to another Replication Server; for example:

- The Replication Server manages primary data that is replicated to databases managed by other Replication Servers.
- The Replication Server manages a database that is a source of replicated stored procedure execution.
- The Replication Server is an intermediate site in an indirect route.

An RSSD does not require a Replication Agent if the replication system has only one Replication Server. Replication Agent for ERSSD is managed by Replication Server. It is automatically started when a route is created. You do not need to decide the ERSSD Replication Agent requirements at installation or configuration time of Replication Server.

See also
- *Database Replication Agent Information* on page 26

Databases that Require Warm Standby

Replication Server can maintain a warm standby database for an Adaptive Server or Oracle database. If the active database fails, clients can switch to the standby database with little interruption.

To set up a warm standby application, you must perform some tasks in Replication Server before and after you add databases to the replication system.

Before you install Replication Server, see *Replication Server Administration Guide Volume 2 > Manage Warm Standby Applications*.

To configure warm standby for Oracle databases, see *Replication Server Heterogeneous Replication Guide > Heterogeneous Warm Standby for Oracle*.

Complete the Replication Server Installation Worksheet

Use the worksheet to record configuration information you need during the Replication Server installation.

The **rs_init** program is used to configure Replication Server software. The sections in the Replication Server Installation Worksheet correspond to the **rs_init** menus, so you can enter information from the worksheet directly into **rs_init**.

rs_init supplies default values for most of the items in the Replication Server installation worksheet and Database setup worksheet. Many of the defaults are constructed from values that you enter. For example, if you enter "TOKYO_RS" for the Replication Server name,

rs_init names the log file TOKYO_RS.log. You can accept the displayed value or enter a different value.

Note: Replication Server reserves identifiers that begin with "rs_". Do not use names that begin with "rs_" for Adaptive Servers, Replication Servers, database objects, or login names. See the *Replication Server Reference Manual* for other restrictions.

Replication Server Installation Worksheet Sample

Make a copy of this worksheet for each Replication Server you install. Complete each section the worksheet as you prepare to install and configure Replication Server.

Keep the worksheet copy to use if you reconfigure your replication system.

Required worksheet items are marked with an asterisk (*). **rs_init** has default values for items that are not marked. To use a default value, leave the unmarked fields blank and fill it in on the worksheet when you run **rs_init**.

Release Directory		
	Release directory:	

Release Server Information		
	Replication Server name:*	
	Is this Replication Server the ID Server? *	Yes	No
	Replication Server error log:	
	Replication Server configuration file:	
	Replication Server password encryption:	Yes	No
	Replication Server character set:	
	Replication Server language:	
	Replication Server sort order:	

Replication Server Security Information		
	(Solaris only) Enable network security?*	Yes	No
	If Yes, security system:	**dce**	**csfkrb5**
	If Yes, login name:	
	If Yes, keytab file name:	

Replication Server Security Information			
	(All platforms) Use Secure Sockets Layer (SSL) security?	Yes	No
	If Yes, SSL identity file:		
	If Yes, SSL private key password (default is password):		

Replication Server Interfaces Information	
	Host name/address:*
	Port:*
	Name alias:*

ID Server Information	
	ID Server name:*
	ID Server user:
	ID Server password:
	Starting Replication Server ID:
	Starting database ID:

Replication Server System Database Choice			
	Will RSSD be embedded? (default is no)	Yes	No

SQL Anywhere Embedded Replication Server System Database Information	
Complete if you selected Yes for "Will RSSD be embedded?"	ERSSD name:*
	ERSSD database file directory:*
	ERSSD transaction log directory:*
	ERSSD backup directory:*
	ERSSD error log directory:*

Adaptive Server Enterprise Replication Server System Database Information	
Complete if you selected No for "Will	RSSD Adaptive Server name:*
	RSSD name:

Adaptive Server Enterprise Replication Server System Database Information			
RSSD be embedded?"	Will RSSD be replicated?*	Yes	No
	Allow HA failover for RSSD connections?	Yes	No
	Create RSSD:*	Yes	No
	SA user:		
	SA password:*		
	Primary user:		
	Primary password:		
	Maintenance login:		
	Maintenance password:		

Adaptive Server Enterprise RSSD Device Information			
Complete if you selected No for "Will RSSD be embedded?"	Size of the RSSD database:		
	RSSD device name:*		
	Create the RSSD device:*	Yes	No
	RSSD device physical name:		
	RSSD device size:		
	Size of the RSSD log:		
	RSSD log device name:*		
	Create the RSSD log device:*	Yes	No
	RSSD log device physical name:		
	RSSD log device size:		

Disk Partition Information			
	Disk partition path:*		
	Logical identifier for disk partition:*		
	Size of disk partition:		
	Start value for partition:		

Remote Site Connections Information	
	Replication Server login name:
	Replication Server password:

RSSD RepAgent Information	
Complete if you selected Yes for "Will RSSD be replicated?"	RS user:
	RS password:

Release Directory

The release directory—also called the installation directory—is where you install Replication Server.

Replication Server creates a new directory structure that allows you to use multiple versions of some components.

See *Installation Directory Structure* in the *Replication Server Installation Guide*.

Verifying Permissions

Verify the "sybase" user has write permission for the Sybase release directory.

See the *Replication Server Installation Guide* for your platform for more information about the "sybase" user.

1. In the .cshrc or .profile file of the "sybase" user, set the SYBASE environment variable to the full path of the Sybase release directory. For example, if your path is /work/sybase, and you use the:

   - C shell – add this line to the .cshrc file:
     ```
     setenv SYBASE /work/sybase
     ```
     At the shell prompt, enter:
     ```
     source .cshrc
     ```
   - Bourne shell – add this line to the .profile file:
     ```
     SYBASE="/work/sybase"; export SYBASE
     ```
     At the shell prompt, enter:
     ```
     . .profile
     ```

2. Record the release directory on the worksheet.

> **Note:** In addition to $SYBASE, Replication Server uses environment variables for Replication Server—$SYBASE_REP, and Open Client/Server™— $SYBASE_OCS. See *Replication Server Installation Guide > Postinstallation Tasks.*

Replication Server Information

Lists the Replication Server information that you need to complete in the corresponding section in the worksheet.

These are the worksheet fields:

Field	Description
Replication Server name	(Required) A name for the Replication Server. The name must be unique in the replication system.
Is this Replication Server the ID Server?	(Required) Select **Yes** if you are installing the ID Server.
Replication Server error log	The path for the Replication Server error log file. Replication Server writes informational and error messages to this text file.
Replication Server configuration file	The path for the Replication Server configuration file, which contains parameters that Replication Server reads at start-up. See *Replication Server Administration Guide Volume 1 > Manage a Replication System.*
Replication Server password encryption	Select **Yes** if you want passwords to be encrypted in the Replication Server configuration file and in the RSSD.
	Warning! If you do not encrypt passwords, anyone with the required permissions can look at the passwords in the configuration files and in the RSSD rs_users system table.
Replication Server character set	The character set that the Replication Server will use. You can specify any Sybase-supported character set that is available for your language.
Replication Server language	The language Replication Server should use to print messages to the error log and to clients. It can be any language that is compatible with the character set you install.
	Additional information about character sets is available in the *Adaptive Server Enterprise Configuration Guide*
Replication Server sort order	The sort order for the Replication Server.

See also
• *ID Server* on page 7

Replication Server Character Sets

The character set used by the Replication Server must match the character set used by the data servers and Replication Agents it controls for replication to work properly.

The character set must be compatible with the character sets of the other Replication Servers in the system. By default, **rs_init** configures Replication Server with the native character set for your platform.

When you create a subscription, Replication Server copies the requested data from the primary database to the replicate database in a process known as subscription materialization. During subscription materialization, the primary data server converts character data to the replicate Replication Server character set. Make sure that the replicate Replication Server character set is installed at the primary data server if it is different from the primary data server character set.

Similarly, when you create a route, make sure that the character set of the destination Replication Server is installed at the data server of the source Replication Server RSSD.

Supported Character Sets

Lists the character sets that Replication Server supports.

The supported character sets in the English language version of Replication Server include:

- cp437 (Code Page 437) – character set used in IBM PCs.
- cp850 (Code Page 850) – IBM/Microsoft Multilingual Character Set, used in IBM PCs.
- deckanji – DEC Kanji Code for JIS-X0208.
- eucgb – EUC GB encoding for Simplified Chinese character sets.
- eucjis – extended UNIX Code for JIS-X0201 and JIS-X0208.
- eucksc – (Code Page 949) EUC KSC Korean encoding.
- gb18030 – GB 18030-2000 standard for Chinese characters, established by the government of the People's Republic of China.
- iso15 – similar to iso_1.
- iso_1 (ISO-8859/1) – 8-bit character set for many systems. This is the default for Adaptive Server with Solaris and IBM RISC System 6000 AIX.
- mac – default Macintosh character set.
- roman8 – Hewlett-Packard character set.
- roman9 – the same as roman8, except for codepoint 0xBA, previously treated as the universal currency symbol, now representing the euro currency symbol.
- sjis (Shift-JIS) – IBM/Microsoft Code for JIS-X0201 and JIS-X02081.
- utf8 – an ASCII-preserving encoding method for Unicode.

The Chinese language includes these character sets:

- eucgb
- gb18030

- utf8

The Japanese language includes these character sets:

- deckanji
- eucjis
- sjis
- utf8

The Korean language includes these character sets:

- eucksc
- utf8

The French, German, and Spanish languages include these character sets:

- cp437
- cp850
- iso15
- iso_1
- mac
- roman8
- roman9
- utf8

Replication Server Language

rs_init configures Replication Server to print messages in U.S. English by default.

Other available languages are Chinese, French, German, Japanese, Korean, and Spanish.
rs_init uses the same identifiers for language names as Adaptive Server. You can select any
language when you install Replication Server as long as the language is compatible with the
character set you install.

If the language for the Replication Server and the language for its connected data servers are
different, the language for the Replication Server must be installed at the data servers. This lets
the data servers return messages to Replication Server in the configured language and
recognize and format dates for the configured language.

For example, the date format "01/02/99" is equivalent to "January 2, 1999" in us_english but
"February 1, 1999" in french. Data servers can format the date correctly only if the Replication
Server configured language is installed.

Replication Server Sort Order

The sort order controls which rows of a table belong in a subscription that has a where clause
involving character data.

The sort order also controls how the identifiers you enter are recognized with regard to case
sensitivity, accents, and so forth.

You can specify any Sybase-supported sort order that is compatible with your character set. For replication to work properly, all sort orders in your replication system should be the same.

By default, **rs_init** configures Replication Server with the binary sort order. Available sort orders vary, depending on the character set you plan to install. Sort orders are located in the Sybase release directory in the directory `charsets/charset_name`, where *charset_name* is the name of the character set.

See the *Replication Server Design Guide* for guidelines on configuring languages, character sets, and sort orders in your replication system.

Replication Server Security Information

Lists the Replication Server security information that you need to complete in the corresponding section in the worksheet.

These are the worksheet fields:

Note: Network security is available only on Solaris

Field	Description
Enable network security	(Required) Select **Yes** to enable external network security.
Type of security system	If you are enabling network security, select DCE or Kerberos.
Login name	The name of the primary user who will make the secure connections. Sybase recommends that you use the name of the Replication Server as the principle user name.
Keytab file name	The full path to the location of the keytab file.
Use SSL security	Select **Yes** if you are using Secure Sockets Layer (SSL) security.
SSL identity file	The full path to the location of the SSL identity file.
SSL private key password	The private key password. The default password is **password**.

See *Replication Server Administration Guide Volume 1 > Manage Replication Server Security* for more information on security.

See also
- *Secure Sockets Layer* on page 91

Replication Server Interfaces Information

Use the Replication Server interfaces information of the worksheet to record information that defines the network port where Replication Server listens for connections. Each entry in the interfaces file is called a listener service.

Note: If you are using network-based security, available with version 11.5 or later, use the directory services of your network security mechanism to register Replication Servers,

Adaptive Servers, and gateway software. See the documentation that comes with your network-security mechanism for details.

These are the worksheet fields:

Field	Description
Hostname/address	(Required) The name of the machine that is running Replication Server. If the machine has more than one host name or IP address, enter the name or address associated with the network you want to use.
Port	(Required) Choose a port number between 1025 and 65535 that is not in use on the machine. Check the `/etc/services` directory or use the **netstat** command to find out which port numbers are in use.
Name alias	(Required) On networks that allow multiple protocols, you can use a name alias to distinguish listener services. An alias is a nickname for a Replication Server. You cannot use the alias for any other server on the network.

ID Server Information

Lists the ID Server information that you need to complete its corresponding section in the worksheet.

These are the worksheet fields:

Field	Description
ID Server name	(Required) If the Replication Server you are installing is the ID Server, **rs_init** assigns an ID Server name identical to this Replication Server name. If you are installing a new Replication Server in an existing Replication Server domain, enter the name of the Replication Server that is acting as the ID Server.
ID Server user	If the Replication Server you are installing is the ID Server, enter the login name that other Replication Servers will use to connect to this ID Server. If you are installing a new Replication Server to an existing Replication Server domain, copy the ID Server user name from the worksheet you completed when you created the ID Server.

Field	Description
ID Server password	If the Replication Server you are installing is the ID Server, enter the password for the ID Server user. If you are adding a new Replication Server to an existing Replication Server domain, copy the ID Server password from the worksheet you completed when you created the ID Server.
Starting Replication Server ID and Starting Database ID	These two configuration parameters allow you to specify separate ranges of Replication Server ID numbers and database ID numbers that an ID Server will assign. Although you are not required to specify ranges, specifying ranges makes it easier to merge replication systems later. **Note:** "Starting Replication Server ID" and "Starting Database ID" in the worksheet apply only to the ID Server. If you are not installing the ID Server, skip to "Replication Server System Database Information".

See also

- *ID Server* on page 7
- *Replication Server System Database Information* on page 20

Default ID Number Ranges and ID Number Allocation for ID Server Domains

An ID Server assigns ID numbers sequentially. Separate ranges of values are maintained for databases and Replication Servers.

Table 1. Database and Replication Server Default ID Number Ranges

Object Type	Minimum ID Number	Maximum ID Number
Database	101	16777215
Replication Server	16777317	33554431

Minimum values for each ID Server must be within the ranges specified. For example, if your organization has three separate ID Server domains, you can set the minimum ID numbers as shown in "Example ID Number Allocation for ID Server Domains."

Table 2. Example ID Number Allocation for ID Server Domains

ID Server Domain	Starting Database ID	Starting Replication Server ID
Finance division	101	16777317
Manufacturing division	100101	16877317

ID Server Domain	Starting Database ID	Starting Replication Server ID
Sales division	200101	16977317

This arrangement provides the ID Server in each domain at least 100,000 ID numbers for databases and 100,000 ID numbers for Replication Servers. You need not set a maximum value, since it is unlikely that you will install more than 100,000 Replication Servers or add more than 100,000 databases to a replication system.

See *Replication Server Administration Guide Volume 1 > Manage a Replication System*, for more information about adding replication system domains.

Replication Server System Database Information

Replication Server maintains its system tables in its system database. You can choose to set up a Replication Server System Database (RSSD) in an Adaptive Server Enterprise database, or an Embedded Replication Server System Database (ERSSD) in a SQL Anywhere (SA) database.

Adaptive Server Enterprise RSSD Information

Lists the Adaptive Server Enterprise RSSD information that you need if you want to have an RSSD.

These are the worksheet fields:

Field	Description
RSSD Adaptive Server name	(Required) The name of the Adaptive Server that will hold the system database for the Replication Server.
RSSD name	The name of the database where the Replication Server system tables will be stored.
Will RSSD be replicated?	(Required) Select **Yes** if the RSSD for the Replication Server requires a Repliction Agent.
Allow HA failover for RSSD connections?	Select **Yes** if the Adaptive Server managing the RSSD supports HA failover and you want RSSD connections to fail over to an alternative Adaptive Server.

Field	Description
Create RSSD	(Required) Select **Yes** if you want **rs_init** to create the system database. Select **No** if the system database already exists or if you want to create it manually. If the database already exists, it should be dedicated to this Replication Server. See the *Replication Server Installation Guide* for disk space and system requirments.
SA user	The login name for the "sa" user for the Adaptive Server that will hold the RSSD. The default is "sa." **rs_init** uses the "sa" login name to create user login names and the system tables in the RSSD.
SA password	(Required) The "sa" password for the Adaptive Server that will hold the RSSD.
Primary user	The login name for the RSSD primary user. Replication Server uses this login name for system table updates. **rs_init** creates the primary user.
Primary password	The password for the RSSD primary user.
Maintenance login	The name of the maintenance user for the RSSD. Replication Server uses this login name to perform operations on the system tables that are replicated from other sites. **rs_init** creates the maintenance user and adds the maintenance user to the RSSD.
Maintenance password	The password for the RSSD maintenance user.

SQL Anywhere ERSSD Information

Lists the SQL Anywhere ERSSD information that you need if you plan to have an ERSSD.

These are the worksheet fields:

Field	Description
ERSSD name	The name of the SQL Anywhere server that will hold the system database for the Replication Server.
ERSSD database file directory	The directory for the SQL Anywhere server database file.
ERSSD transaction log directory	The directory for the SA server transaction log.
ERSSD backup directory	The directory for the SQL Anywhere server backup files.

Field	Description
ERSSD error log directory	The directory for the SQL Anywhere server error log file.

Note: rs_init automatically fills these fields with default values on the same device. However for better performance and recoverability, place the ERSSD database directory, ERSSD transaction log directory, and ERSSD backup directory on separate devices.

See *Replication Server Administration Guide Volume 1 > Manage a Replication System* for more information on ERSSDs.

Adaptive Server Enterprise RSSD Device Information

Complete the "RSSD Device Information" section of the worksheet if you want **rs_init** to create the RSSD and you selected Yes for "Create RSSD".

However, if you do not want **rs_init** to create the RSSD, skip this section and go to "Disk partition information" to complete its corresponding section in the worksheet.

These are the worksheet fields:

Field	Description
Size of the RSSD database	The size, in megabytes, of the data portion.
RSSD device name	(Required) The name of the Adaptive Server logical device where you want to create the RSSD. If the device already exists, it must have enough space available to create the new database.
Create the RSSD device	(Required) Select **Yes** if you want **rs_init** to create the RSSD data device on the Adaptive Server database device. Select **No** if the RSSD data device already exists, or if you will create it before you run **rs_init**.
RSSD device physical name	If you want **rs_init** to create the RSSD device, enter the physical device name for the database device on which you want to store the RSSD. If you do not want **rs_init** to create the RSSD data device, leave this field blank.
RSSD device size	If you want **rs_init** to create the RSSD device, enter the capacity, in megabytes, of the physical device. The device size must be at least the size you specified for the data portion of the RSSD database. If you do not want **rs_init** to create the RSSD device, leave this entry blank.

Field	Description
Size of the RSSD log	The size, in megabytes, for the RSSD database log.
	Note: Sybase recommends that you store the RSSD log on a device other than the master device, and on a different device from the RSSD data.
RSSD log device name	(Required) The logical name for the device you want **rs_init** to use for the RSSD log.
Create the RSSD log device	(Required) Select **Yes** if you want **rs_init** to create the RSSD log device in Adaptive Server.
	Select **No** if the device already exists or if you will create it before you run **rs_init**.
RSSD log device physical name	If you want **rs_init** to create the RSSD log device, enter the physical device name for the device.
	If you do not want **rs_init** to create the RSSD log device, or if you are using a single device for both the RSSD and its log (this is not recommended), leave this field blank.
RSSD log device size	If you want **rs_init** to create the RSSD log device, enter the capacity, in megabytes, of the physical device. The device size must be at least the size you specified for the RSSD log.
	If you do not want **rs_init** to create the RSSD log device, or if you are using a single device for both the RSSD and its log (this is not recommended), leave this field blank.

RSSD Device
The RSSD device is the Adaptive Server database device where **rs_init** creates the Replication Server RSSD.

Therefore, choose a database device other than the default master device, that has the minimum amount of space required. Use **sp_helpdevice** to list the available devices.

Note: rs_init by default uses a minimum of 40MB for the database size, and 16MB for the RSSD log size.

Select one of the following if the Adaptive Server does not have a device available for the RSSD:

- Add a device and record the information on your worksheet
- Have **rs_init** create the device when it installs Replication Server

Note: If the Adaptive Server is not on the local machine, you must create the device yourself. **rs_init** incorrectly evaluates the local machine's file system instead of the remote

machine's file system when it validates the path name and disk space requirements of the device.

If you want **rs_init** to create the device, verify that the disk file or raw partition you want to use exists and has the proper permissions . Execute **sp_configure** devices in Adaptive Server to verify that the value for devices is high enough to allow you to add a new device. See the *Adaptive Server Enterprise System Administration Guide* for information about device requirements.

Disk Partition Information

Lists the disk partition information that you need to complete the corresponding section in the worksheet.

Replication Server uses disk partitions for stable queues, which temporarily store the data it receives and sends.

Replication Server can use raw partitions, if they are available on your platform, or operating system files. Raw partitions provide improved recovery capability because the operating system does not buffer their I/O.

The minimum partition size is 20MB. You can add more partitions after the Replication Server is installed.

Field	Description
Disk partition path	(Required) The name of the physical disk device or the full path name of a file to be used for the disk partition. If you use an operating system file, there must be enough space on the file system for Replication Server to extend the file to the size you specify. You must have already created the operating system file.
Logical identifier for disk partition	(Required) A logical identifier for the partition. This name is used in commands and in Replication Server messages to identify the disk partition. See *Replication Server Reference Manual > Topics > Identifiers*.
Size of disk partition	The size, in megabytes, of the disk partition. The partition must be at least 20MB. See *Replication Server Administration Guide Volume 1 > Replication Server Technical Overview > Transaction Handling with Replication Server > Stable Queues > Partitions for Stable Queues* and *Replication Server Design Guide > Capacity Planning*.

Field	Description
Start value for partition	This value prevents Replication Server from using the beginning of a raw disk partition. It is required if your operating system stores information, such as configuration maps, at the beginning of the disk partition. Enter the number of megabytes Replication Server should ignore. For example, if you enter 1, Replication Server will not use the first megabyte of the raw disk partition.

Creating a Disk Partition

Create raw partitions in Replication Server if they are available on your platform or operating system files. Raw partitions provide improved recovery capability because the operating system does not buffer their I/O.

Prerequisites

Verify that the operating sytem file that you will use for a partition exist before you specify it in **rs_init**.

Task

1. To create an empty file for a partition, use a text editor or the UNIX touch command. Verify that there is enough free space on the file system for the file to grow to the size you specify. Replication Server extends the file to the full partition size during installation. Grant read and write permissions on the file to the "sybase" user.

   For example, log in as the "sybase" user, and create the file with this command, where path is the full *path* to the disk file:

   ```
   touch path
   ```

2. Set the permissions on the file:

   ```
   chmod 600 path
   ```

See also
• *Entering Disk Partition Information* on page 50

Remote Site Connections Information

The Remote site connections information section of the worksheet defines a login name for other Replication Servers to use when logging in to this Replication Server.

These are the worksheet fields:

Field	Description
Replication Server login name	The login name that other Replication Servers will use to connect to this Replication Server.
Replication Server password	Replication Server password

Database Replication Agent Information

Lists the database Replication Agent information that you need to complete the corresponding section in the worksheet.

These are the worksheet fields:

Field	Description
RS user	The login name that the RSSD Replication Agent will use to connect to the Replication Server
RS password	The password for the RS user login name.

See also
- *Databases that Require Replication Agents* on page 8

Complete the Database Setup Worksheet

Use the worksheet to record configuration information you need for each database you plan to add to the replication system. Refer to the information on the worksheet when you use **rs_init** to add the database to your replication system.

Each primary or replicate database in your replication system is managed by a Replication Server. The database may also require a Replication Agent.

See the *Replication Server Design Guide* for guidelines on configuring character sets, languages, and sort orders in your replication system.

Database Setup Worksheet Sample

Make a copy of this worksheet for each database you add to the replication system. Complete the worksheet as you prepare to install and configure Replication Server.

Keep the worksheet copy to use if you reconfigure your replication system.

Required worksheet items are marked with an asterisk (*). **rs_init** has default values for items that are not marked. To use the default values, leave the unmarked field blank and fill it on the worksheet when you run **rs_init**.

Replication Server Information			
	Replication Server name:*		
	RS SA user:*		
	RS SA password:*		

Replication Server Interfaces Information			
	Host name/address:*		
	Port:*		
	Name alias:*		

Database Information			
	Adaptive Server name:*		
	SA user:		
	SA password:		
	Database name:		
	Will the database be replicated?*	Yes	No
	Maintenance user:		
	Maintenance password:		
	Is this a physical connection for an existing logical connection?	Yes	No

Logical Connection Information			
Complete if you selected Yes for "Is this a physical connection for an existing logical connection?"			
	Is this an active connection or a standby connection?*	Active	Standby
	Logical DS name:*		
	Logical DB name:*		
Complete the rest of the items in this section only if you selected "Standby" in response to "Is this an active connection or standby connection?"			
	Active DS name:*		

	Active DB name:*		
	Active DB SA user:*		
	Active DB SA password:*		
	Initialize standby using **dump** and **load**? *	Yes	No
	Use dump marker to start replicating to standby?*	Yes	No

Database RepAgent Information
Complete the following if you selected Yes for "Will the database be replicated?"

	RS user:	
	RS password:	

Replication Server Information

The Database Setup Worksheet includes Replication Server information needed to identify the Replication Server that will manage the database.

The Replication Server must be installed and running before you can add the database to the replication system.

These are the worksheet fields:

Field	Description
Replication Server name	(Required) The name of the Replication Server that will manage the database.
RS SA user	(Required) The Replication Server System Administrator (RS SA) login name (usually "sa").
RS SA password	(Required) The password for the Replication Server SA user.

Replication Server Interfaces Information

Copy the information you have provided in the Replication Server interfaces information section of the Replication Server installation worksheet for the Replication Server that will manage this database. Since the Replication Server is already installed, you do not have to enter this information again unless you start **rs_init** in a release directory that uses a different interfaces file.

Note: If you are using network-based security, available with version 11.5 or later, use the directory services of your network security mechanism to register Replication Servers,

Adaptive Servers, and gateway software. See the documentation that comes with your network-security mechanism for details.

These are the worksheet fields:

Field	Description
Hostname/address	(Required) The name of the machine that is running Replication Server. If the machine has more than one host name or IP address, enter the name or address associated with the network you want to use.
Port	(Required) Choose a port number between 1025 and 65535 that is not in use on the machine. Check the /etc/services directory or use the **netstat** command to find out which port numbers are in use.
Name alias	(Required) On networks that allow multiple protocols, you can use a name alias to distinguish listener services. An alias is a nickname for a Replication Server. You cannot use the alias for any other server on the network.

Database Information

The Database Setup Worksheet includes database information that identifies the database you are adding to the replication system.

Field	Description
Adaptive Server name	(Required) The name of the Adaptive Server where the database resides.
SA user	The "sa" login name for the Adaptive Server managing the database.
SA password	The password for the SA user login name.
Database name	The name of the database.
Will the database be replicated?	(Required) Select **Yes** if the database requires a Replication Agent.
Maintenance user	A login name for Replication Server to use when updating replicated data. This user must be granted all permissions necessary to maintain the replicated data. **rs_init** creates this login, if it does not already exist, and grants the necessary permissions.
	See the *Replication Server Administration Guide Volume 1* for more information about login names for replication system components.

Field	Description
Maintenance password	The password for the maintenance user.
Is this a physical connection for an existing logical connection?	Select **Yes** if you are adding an active or standby database for a warm standby application, and **No** if you are not. See the instructions for setting up warm standby databases in *Replication Server Administration Guide Volume 2 > Managing Warm Standby Applications*

See also
- *Databases that Require Replication Agents* on page 8

Logical Connection Information

The Database Setup Worksheet includes logical connection needed if you are adding a database that is part of a warm standby database application.

Setting up a warm standby application requires that you perform additional tasks in the Replication Server before and after you use **rs_init** to add the databases to the replication system.

To configure warm standby for Adaptive Server databases, see *Replication Server Administration Guide Volume 2 > Manage Warm Standby Applications*. To configure warm standby for Oracle databases, see *Replication Server Heterogeneous Replication Guide > Heterogeneous Warm Standby for Oracle*.

These are the worksheet fields:

Field	Description
Is this an active connection or standby connection?	(Required) Select **Active** if you are adding the active database for the warm standby application. Select **Standby** if you are adding the standby database for the warm standby application.
Logical DS name	(Required) The name of the logical data server for the logical connection. You must have already created it using **create logical connection**.
Logical DB name	(Required) the name of the logical database for the logical connection. You must have already created it using **create logical connection**. Complete the rest of the items in this section only if you have selected a standby connection over an active connection.

Field	Description
Active DS name	(Required) The name of the data server with the active database. Enter the actual data server name, not the logical data server name defined for the warm standby database pair.
Active DB name	(Required) The name of the active database. Enter the actual database name, not the logical database name defined for the warm standby database pair.
Active DB SA user	(Required) The login name of the system administrator for the data server of the active database. Replication Server uses this login name to set up warm standby applications.
Active DB SA password	(Required) The password for the system administrator login name of the data server of the active database.
Initialize standby using dump and load?	(Required) Select **Yes** if you plan to initialize the standby database with the contents of the active database with the **dump** and **load** commands. Select **No** if you plan to initialize the standby database with the **bcp** utility, or if you do not need to initialize the standby database.
Use dump marker to start replicating to standby?	(Required) Select **Yes** if transactions will be executed in the active database while the standby database is initialized.

Database RepAgent Information

The Database Setup Worksheet includes database RepAgent information needed if the database requires a Replication Agent and you have selected to replicate the database.

These are the worksheet fields:

Field	Description
RS user	The Replication Server login name that the Replication Agent will use to connect to the Replication Server. The default Replication Server user login name and password for this Replication Agent is the default login name and password for the Replication Server user for the RSSD RepAgent, which you noted in the "RSSD RepAgent" section of the Replication Server installation worksheet. If you want this user to use a different login name and password, you must first create the user in the Replication Server and then grant connect source permission to the user.
RS password	The password for the RS user login name.

Configure Replication Server and Add Databases Using rs_init

Use **rs_init** to configure Replication Server and to add databases to your replication system. You can use **rs_init** interactively or with a resource file.

rs_init in Interactive Mode

Using **rs_init** in interactive mode is easy to use and catches mistakes that might be missed when using **rs_init** with a resource file.

rs_init in interactive mode provides command keys and menu prompts to help you make selections and move between windows. It also rejects invalid entries and displays warnings or error messages when you make improper selections. If you make a mistake, you can change your entry and continue with the installation session.

Starting rs_init

Run **rs_init** and provide the information from the Replication Server installation worksheet.

Each section of the worksheet corresponds to a window or a menu in **rs_init**. You can also start **rs_init** with other command line options for your replication environment.

1. Log in to your machine as the "sybase" user.

   Note: You must run **rs_init** on the machine on which the Replication Server will run.

2. Set the SYBASE environment variable to the path of the Sybase release directory, as described in the *Replication Server Installation Guide*.

3. Change to the Sybase release directory:
   ```
   cd $SYBASE/$SYBASE_REP
   ```

4. Enter:
   ```
   $SYBASE/$SYBASE_REP/install/rs_init
   ```

5. Choose Configure a Server Product, then Replication Server.

   From the Configure Replication System menu you can:
   - Configure a new Replication Server.
   - Add a database to the replication System.
   - Upgrade or downgrade Replication Server.
   - Enable password encryption for a Replication Server or alter passwords in configuration files.

See also
- *Password Encryption* on page 87
- *rs_init Command Line Options* on page 44
- *Configuring a New Replication Server* on page 46
- *Adding a Database to the Replication System* on page 51

rs_init Interactive Command Keys

Lists the command keys that you can use in interactive mode

Keys	Command	Action
Ctrl+A	Accept	Accept the values currently listed in the menu and move to the next menu.
Ctrl+B	Backward	Scroll backward one page. Use this command key when a menu or list does not fit on a single screen.
Ctrl+C	Quit	Quit **rs_init** and return to the shell prompt. Your terminal may use a key mapping other than Ctrl+C. Use the key sequence that is mapped to the SIGINT signal.
Ctrl+F	Forward	Scroll forward one page. Use this command key when a menu or list does not fit on a single screen.
Ctrl+R	Refresh	Refresh the screen.
Ctrl+X or Esc	Exit	Exit the current menu or prompt.
Ctrl+W	Write to re-source file	Generate a file containing the values specified during the current **rs_init** session. The default file name is `$SYBASE/$SYB-ASE_REP/init/logs/resource.dmp`. **rs_init** prompts you to enter an alternative file name. You can copy information from this generated file into one of the resource file templates.
?	Help	Display an online help screen.

rs_init With a Resource File

Using **rs_init** with resource files to install, configure, upgrade, or downgrade may save time if you plan to install multiple Replication Servers and many of the **rs_init** values are similar.

A resource file is an ASCII-formatted file that contains configuration information for **rs_init**. Instead of entering configuration variables during an interactive **rs_init** session, use any text editor to edit the resource file to specify the variables, then specify the resource file name on the **rs_init** command line.

Using a Resource file for rs_init

Prepare and use a resource file for **rs_init**.

Sybase supplies several resource file templates for Replication Server. Each template contains the attributes that are valid for a certain **rs_init** operation. Copy and edit a template file, which is supplied with the Replication Server software.

1. Select the resource file template you want to use.
2. Make a copy of the template and rename it to distinguish it from the original. For example:
   ```
   cp install.rs install.rsx
   ```

   Note: You must have read and write permissions on the copy of the resource file template that you want to edit. You may need to grant write permission to yourself as the file owner.
3. Use a text editor to edit the new resource file.

   To use the **vi** editor, for example, enter:
   ```
   vi install.rsx
   ```
4. When you have finished editing the resource file, start **rs_init** from the operating system prompt, using the **-r** flag to specify your resource file. For example:
   ```
   $SYBASE/$SYBASE_REP/install/rs_init -r install.rsx
   ```

   Warning! If you try to start a resource file session with an input file that is not an edited copy of one of the resource file templates, **rs_init** fails upon validation of the input file.

   You can also start **rs_init** using a resource file with other command line options for your replication environment.

See also
- *rs_init Command Line Options* on page 44

Replication Server Resource File Templates
The resource file templates for Replication Server are located in subdirectories under the `init` subdirectory of the Sybase installation directory.

Template Name	Description
install.rs	Configures a new Replication Server.
setupdb.rs	Adds a database to the replication system.
upgr.rs	Upgrades an RSSD for a Replication Server. Use this template after installing new Replication Server software.
upgrdb.rs	Upgrades a user database that is already part of a replication system.

Template Name	Description
downgr.rs	Downgrades an RSSD for a Replication Server. Use this template before reinstalling the previous Replication Server software release.
eers.rs	Enables password encryption for a Replication Server.
updrscf.rs	Alters a password in a Replication Server configuration file.

Syntax and Parameters for a Resource File
Resource files list the attributes and their values to be defined during an **rs_init** session.

Syntax

Attribute entries are in this format:

```
product_name.attribute:value
```

Parameters

- **product_name –** Either sybinit or rs.
- **attribute –** An item for which you are supplying information, such as a server name or network protocol.
- **value –** The value you want to assign to the attribute.

Usage

- Resource file entries can be only one line long (80 characters).
- **rs_init** ignores blank lines and lines that begin with a pound sign (#).
- If you do not enter a valid value for every required attribute, **rs_init** returns an error when you use the resource file.
- If a Sybase-defined default value exists for an attribute, you can specify the default with the special value USE_DEFAULT.
- Certain resource file attributes can take lists of values. Attributes that can accept lists of values end with "_list." Use commas to separate values. For example:
  ```
  rs.rs_rs_network_protocol_list:tcp,ipx
  ```
- If you use a resource file to upgrade or modify a Replication Server, you can use the UNCHANGED value to indicate that a particular attribute should remain unchanged.
- You cannot use the USE_DEFAULT or the UNCHANGED value for required attributes with a null default value and you cannot enter USE_DEFAULT or UNCHANGED for the passwords that you specify with these attributes:

Password Attribute	User ID
rs.rs_idserver_pass	ID Server user
rs.rs_rs_sa_pass	Replication Server sa login
rs.rs_rssd_prim_pass	RSSD primary user
rs.rs_db_maint_password	Database maintenance user
rs.rs_rssd_maint_pass	RSSD maintenance user
rs.rs_ltm_rs_pass	Log transfer manager login ID
rs.rs_rs_pass	Replication Server login name

You must provide passwords for these attributes. If the attribute has no value or is not in the resource file, you see a message in the **stdout**. For example, if you do not provide a value for the rs_db_maint_password attribute in the setupdb.rs resource file, you see:

```
Attribute 'rs_db_maint_password' could not be set because '' is an
invalid value
```

To ensure security, you must enter a password that complies with the password security requirements such as a minimum password length that administrator has enforced. See *Replication Server Administration Guide Volume 1 > Manage Replication Server Security > Managing Replication Server User Security > Manage Replication Server Login Names and Passwords > Password Configuration Options for All Users.*

• If you are rebuilding an existing Replication Server, you must drop the RSSD or ERSSD before you start **rs_init**.

Sample Replication Server Resource File
See a sample resource file. Your resource file will have different values for many of the attributes, but will be similar to the sample.

```
 ########################################################

# RESOURCE TEMPLATE
#
# This file contains a list of configuration attributes
# for Sybase Replication Server. This is the template
# for configuring a new Replication Server. DO NOT EDIT
# THIS FILE.

# Copy the template to another file and edit that.
#
# Syntax of the resource file is:
#
#  product_name.attribute:value
#
# Attributes with a value of USE_DEFAULT will use the
#Sybase defaults for this platform.
#
#NOTES:
```

```
# Generic attributes are prefaced by "sybinit." The
# only generic attributes are "release_directory" and
# "product."
#
########################################################

########################################################
# RELEASE LOCATION
#
sybinit.release_directory: /usr/u/sybase

# The product that you wish to configure. Possible
values are:
# rs
#
sybinit.product: rs

########################################################
# REPLICATION SERVER ATTRIBUTES
#
# This operation installs a new Replication Server.
rs.rs_operation: rs_install

########################################################
# ID SERVER INFORMATION
#
# Name of the ID Server
rs.rs_idserver_name: IDRS
# Is the ID Server the same as the Replication Server
that is being installed ? Default is no
rs.rs_id_server_is_rs_server: yes
# Login that other Replication Servers will use to
# connect with the ID Server. Default is
#<rs_idserver_name>_id_user.
rs.rs_idserver_user: USE_DEFAULT

# Password for the ID Server user. This is a required field.
rs.rs_idserver_pass:
# The next two attributes should be set only when
#installing an ID Server and there are multiple ID
#Server domains
# First ID used for Replication Servers in this ID
#Server domain

rs.rs_start_rs_id: USE_DEFAULT

# First ID used for databases in this ID Server domain

rs.rs_start_db_id: USE_DEFAULT

########################################################
# REPLICATION SERVER INFORMATION
#
# Replication Server name
rs.rs_name: NYRS
```

```
# Replication Server sa password. This is a required field.
rs.rs_rs_sa_pass:
# Will the Replication Server manage databases with
# primary data, submit asynchronous transactions, or
#serve as an intermediate site in an indirect route ?
#Default is no
rs.rs_requires_ltm: no

# Will the Replication Server(11.0 or higher) manage
#databases with primary data, submit asynchronous
#transactions, or serve as an intermediate site in an
#indirect route ? Default is yes
rs.rs_needs_repagent: yes

# Locations of the errorlog and config file for the
# Replication Server.
# The default names of these files are <rs_name>.log and
#<rs_name>.cfg respectively.
# The default directory in which these files are located
#is the current working directory on Unix platforms, and
#in %SYBASE%\install on PC platforms.

rs.rs_rs_errorlog: USE_DEFAULT
rs.rs_rs_cfg_file: USE_DEFAULT

# Character set to be used by this Replication Server
#(and the RSSD LTM if needed)
rs.rs_charset: USE_DEFAULT
# Language to be used by this Replication Server (and
#the RSSD LTM if needed)
rs.rs_language: USE_DEFAULT

# Sort order to be used by this Replication Server (and
#the RSSD LTM if needed)
rs.rs_sortorder: USE_DEFAULT

#######################################################
# REPLICATION SERVER SYSTEM DATABASE CHOICE
#
# Is this an embedded rssd
rs.rs_rssd_embedded: no
#######################################################
# EMBEDDED REPLICATION SERVER SYSTEM DATABASE
#INFORMATION
# The following only applies if embedded RSSD is chosen
#
# erssd name
rs.rs_erssd_name: rep_erssd
# embedded rssd database directory
rs.rs_erssd_database_dir: /work/database
# embedded rssd transaction log directory
rs.rs_erssd_translog_dir: /work/translog
# embedded rssd backup directory
rs.rs_erssd_backup_dir: /work/backup
# embedded rssd error log directory
rs.rs_erssd_errorlog_dir: /work/errorlog
```

```
############################################################
# REPLICATION SERVER SYSTEM DATABASE INFORMATION
# The following only applies if embedded RSSD is not
#chosen
# Name of the Adaptive Server that will manage the
# Replication Server's system database.
rs.rs_rssd_sqlsrvr: NYDS

# Name of the database where the Replication Server
#system tables will be stored. Default is <rs_name>_RSSD
rs.rs_rssd_db: USE_DEFAULT
# Do you want the RSSD connections to allow HA failover?
#Default is no
rs.rs_rssd_ha_failover: no
# Do you want rs_init to create the system database ?
#Default is no
rs.rs_create_rssd: no
# sa login for the system Adaptive Server - default is sa
rs.rs_rssd_sa_login: USE_DEFAULT

# sa password for the system Adaptive Server
rs.rs_rssd_sa_pass:
# Name of the RSSD primary user. Default is
#<rs_rssd_db>_prim
rs.rs_rssd_prim_user: USE_DEFAULT


# Password for the RSSD primary user. This is a required field.
rs.rs_rssd_prim_pass:
# Name of the RSSD maintenance user. Default is
#<rs_rssd_db>_maint
rs.rs_rssd_maint_user: USE_DEFAULT


# Password for the RSSD maintenance user.  This is a required field.
rs.rs_rssd_maint_pass:

# The dbo_user and dbo_password attributes are not used
#by default. They should be used only if the RSSD
#requires an LTM and the log should be scanned by
#someone other than rs_rssd_sa_login. This user should
already exist in the database.

# Name of the Database Owner for the RSSD
rs.rs_rssd_dbo_user: USE_DEFAULT

# Password for the database owner
rs.rs_rssd_dbo_pass:

############################################################
# REPLICATION SERVER SYSTEM DATABASE DEVICE INFORMAITON

# The following only applies if embedded RSSD is not
# chosen (USED ONLY IF RS_INIT IS TO CREATE THE RSSD)
```

```
# Size of the system database in MB. Default and minimum
# is 40
rs.rs_rsdddb_size: 40

# Size of the log for the system databas in MB. Default
#and minimum is 32
rs.rs_rssd_log_size:32

# Name of the device on which the system database is to
#be created Default is master
rs.rs_rssd_db_device_name: dbdev
# Do you want rs_init to create this device for the
#system database ?
# Default is no
rs.rs_create_rssd_database_dev: no
# Physical pathname of the device for the system
#database
rs.rs_rssd_db_device_path: /work/dev1
# Size of the device for the system database
rs.rs_rssddb_device_size: 40
# Name of the device on which the log for the system
#database is to be created
rs.rs_rssd_log_device_name: logdev
# Do you want rs_init to create this device for the log
# for the system database ? Default is no

rs.rs_create_rssd_log_dev: no
# Physical pathname of the device for the log for the
# system database
rs.rs_rssd_log_device_path: /work/dev2
# Size of the device for the log for the system database
rs.rs_rssd_log_device_size:32
########################################################
# DISK PARTITION INFORMATION
#
# Full path name of a raw disk partition for the
# Replication Server
rs.rs_diskp_name: /work/dev3
# Logical identifier name for the raw disk partition for
# the Replication Server
rs.rs_diskp_lname: part1
# Size, in megabytes, of the raw disk partition.Default
# is 20.
rs.rs_diskp_size: 20
# The offset, in megabytes, at which the Replication
#Server should begin writing in the raw disk partition.
#Default is 0
rs.rs_diskp_vstart: 0
#######################################################
# REMOTE SITE CONNECTION INFORMATION
#
# Replication Server login name that other Replication
# Servers will use to connect with this Replication
# Server Default is <rs_name>_rsi
rs.rs_rs_user: USE_DEFAULT
```

```
# Password for the Replication Server login name. This is a required
field.
rs.rs_rs_pass:
########################################################
# SYSTEM DATABASE LOG TRANSFER MANAGER INFORMATION
#
# (IF RSSD NEEDS LTM)
# Name of the RSSD LTM. Default is
# <rs_rssd_sqlsrvr>_<rs_name>_RSSD_ltm.
rs.ltm_name: NY_LTM
# Replication Server login name that the log transfer
# manager will use when connecting to the Replication
# Server
# Default is <rs_name>_ltm
rs.rs_ltm_rs_user: USE_DEFAULT


# Password for the login name for the log transfer
# manager.
# This is a required field.
rs.rs_ltm_rs_pass:
# Login name for the user who will start and shutdown
#the log
# transfer manager for the Replication Server system
# database
# Default is sa
rs.rs_ltm_admin_user: USE_DEFAULT

# Password for the admin user.This is a required field.
rs.rs_ltm_admin_pass:
########################################################
# ID SERVER INTERFACES INFORMATION

# These attributes are valid only for Unix platforms.
# On PC platforms, adding interface file entries through
# resource files is not supported.
# rs.do_add_id_server must be no on these platforms.
#
# Add ID Server to interfaces file? Default is no
rs.do_add_id_server: USE_DEFAULT

# Connect retry count; number of times client tries to
# connect to ID Server before giving up
rs.rs_id_server_connect_retry_count: USE_DEFAULT
# Connect retry delay time (in seconds); amount of time
# client waits between each connection attempt
rs.rs_id_server_connect_retry_delay_time: USE_DEFAULT
# Notes associated with ID Server interfaces file entry
rs.rs_id_server_notes: Default Sybase Configuration
# Protocol for ID Server network listener
rs.rs_id_server_network_protocol_list: tcp
# Name of host for ID Server
rs.rs_idserver_hostname: herbie
# Port numbers for network listener
```

```
rs.rs_idserver_port: 5002
#########################################################
# REPLICATION SERVER INTERFACES INFORMATION

# These attributes are valid only for Unix platforms.
# On PC platforms,adding interface file entries through
# resource files is not supported.
# rs.do_add_replication_server must be no on these
# platforms.
#
# Add Replication Server to interfaces file?
rs.do_add_replication_server: no
# Connect retry count; number of times client tries to
# connect
# to Replication Server before giving up
rs.rs_rs_connect_retry_count: USE_DEFAULT
# Connect retry delay time (in seconds); amount of time
# client waits between each connection attempt
rs.rs_rs_connect_retry_delay_time: USE_DEFAULT
# Notes associated with Replication Server interfaces
# file entry
rs.rs_rs_notes: Default Sybase Configuration
# Protocol for Replication Server network listener
rs.rs_rs_network_protocol_list: tcp
# Name of host for Replication Server
rs.rs_rs_hostname: herbie
# Port numbers for network listener
rs.rs_rs_port: 5005
#########################################################
# LOG TRANSFER MANAGER INTERFACES INFORMATION - IF RSSD
# HAS LTM

# These attributes are valid only for Unix platforms.
# On PC platforms,adding interface file entries through
# resource files is not supported.
# rs.do_add_ltm must be no on these platforms.

# Add Log Transfer Manager to interfaces file?
rs.do_add_ltm: no

# Connect retry count; number of times client tries to
# connect to Log Transfer Manager before giving up
rs.rs_ltm_connect_retry_count: USE_DEFAULT
# Connect retry delay time (in seconds); amount of time
#client waits between each connection attempt
rs.rs_ltm_connect_retry_delay_time: USE_DEFAULT
# Notes associated with Log Transfer Manager interfaces
# file entry
rs.rs_ltm_notes: Default Sybase Configuration
# Protocol for Log Transfer Manager network listener
rs.rs_ltm_network_protocol_list: tcp
# Name of host for Log Transfer Manager
rs.rs_ltm_hostname: herbie
# Port numbers for network listener
rs.rs_ltm_port: 5000
#########################################################
```

```
# REPLICATION SERVER SECURITY INFORMATION
# These attriibutes apply to the security features
# available for the replication server. This option is
# only available on Solaris and NT.
# Enable external network security
rs.rs_network_security_enable: no
# Type of network security for the Replication Server,
# choices are "dce" or "csfkrb5",
rs.rs_network_security_system: USE_DEFAULT
# Login name for the user who will principle user, this
# login name will be used for all secure connections,
# Sybase recommends the name of the Replication Server
# as the principle user name.
rs.rs_principal_user_name: USE_DEFAULT
# Full path to the location of keytab file
rs.rs_keytab_file: USE_DEFAULT
# Use Secure Socket Layer(ssl) security
rs.rs_use_ssl: no
# Full path to the location of the ssl identity file
rs.rs_ssl_identity_file: USE_DEFAULT

# Password for the ssl private key
rs.rs_ssl_pkey_password:
# end of resource file
```

rs_init Command Line Options

Familiarize yourself with the available **rs_init** command line options for a resource file. You can use some of these options in an interactive **rs_init** session.

You can also start **rs_init** with other command line options for your replication environment:

```
rs_init -r resource_file_name [-option] [parameter]
```

where:

- *option* – is the letter specifying a command line option.
- *parameter* – is any valid parameter for that option.

You can specify more than one command line option. For a resource file session, you must specify the **-r** option and the resource file name.

Flags	Parameters	Explanation
-a	none	Validates the resource file and exits without performing the configuration task. Without the -**a** flag, **rs_init** validates the resource file and continues the product configuration. You must use the -**r** flag with the -**a** flag.

Flags	Parameters	Explanation
-c	See the list of supported character sets.	Specifies the character set used for all messages and prompts. The -**c** flag forces **rs_init** to use a particular character set. Otherwise, **rs_init** uses your system's default character set.
-e	none	Records information about the user's environment and exits. Information is written to the log file. Without the -**e** flag, **rs_init** records user environment information and continues the product configuration.
-h	none	Prints out usage message (help) and exits.
-l	French, German, Japanese, Spanish, Chinese, Korean	Specifies the language used for all messages and prompts. The -**l** flag forces **rs_init** to use a particular language. Without this option, **rs_init** uses U.S. English.
-log	Path and file name of log file	Lets the user specify location of the session log file. Without the -**log** flag, **rs_init** uses the default log location ($SYBASE/init/log).
-r	Path and file name of resource file	Specifies that **rs_init** is to take input from the listed resource file. Use this flag in noninteractive sessions to configure new server products, upgrade existing server products, or modify existing server products.
-s	Path of Sybase release directory	Specifies the Sybase release directory. If you do not use the -**s** flag, **rs_init** assumes that the Sybase release directory is the current directory.
-T IGNORE_WARNINGS	none	Entering certain values for some attributes causes **rs_init** to exit with a warning. (**rs_init** exits, for instance, if you specify an operating system file for the location of the master device.) Invoking an **rs_init** resource file session with the -**T IGNORE_WARNINGS** flag allows the **rs_init** session to continue.
-v	none	Prints the **rs_init** version string and exits.

See also
* *Supported Character Sets* on page 15

Configuring a New Replication Server

Configure a new Replication Server in your replication environment using **rs_init**.

Prerequisites
Complete the Replication Server installation worksheet.

Task

1. From the **rs_init** menu, select **Configure a Server Product**. The screen you see lists the Sybase products that are installed in the release directory.
2. Select **Replication Server**.
3. Select **Install a New Replication Server**.
   You see a list of tasks you must complete to install a new Replication Server.

   Note: The Install a New Replication Server option only lets you configure a new Replication Server; it does not allow you to install additional software.

   Select each task and complete the windows until the status of each task is "Complete."

Entering Replication Server Information

Enter the configuration information for the Replication Server using **rs_init**.

1. In the New Replication Server window, select Replication Server Information.
   You see the Replication Server Name window.
2. Enter the name you recorded in the "Replication Server information" section of your worksheet, and press Return. Press Ctrl+A to accept the name.
3. Select each item on the Replication Server Information screen and enter the value you recorded on your worksheet. If you use a default value, record it on your worksheet.

   Note: For character sets, language and sort order, enter the appropriate number for the value you want to select from the list displayed.

   See *Adaptive Server Enterprise Configuration Guide > Customizing Localization for Adaptive Server* for more informarion on the character sets and sort orders.

   In some configurations, if you set Replication Server language before you set its character set, you receive an error message. To avoid this, set the character set before you set the language.

4. Press Ctrl+A to accept the information and return to the New Replication Server window.

Editing the Interfaces File

Modify the interfaces file. The interfaces file contains network address information for each Adaptive Server and Replication Server in your replication system.

1. To reach the `interfaces` file window, select Replication Server Interfaces Information from the Replication Server Information window

2. Select **dsedit** to edit the `interfaces` file if you are using an **xterm** window, or from the command line you can execute the **dscp** utility to edit the `interfaces` file if **xterm** is not available. **dscp** is in `$SYBASE/$SYBASE_OCS/bin`.

Note: Sybase recommends that if you are using Replication Server with network-based security, you use the directory services of your network security mechanism to register Replication Servers, Adaptive Servers, and gateway software. See the documentation that comes with your network security mechanism for details.

See the *Open Client and Open Server Configuration Guide* for more information on **dsedit**.

Entering ID Server Information

Complete the ID Server Information dialog box.

Prerequisites

- Ensure that you have the login name and password (ID Server User/Password) that all Replication Servers use to log in to the ID Server when you configure the ID Server.
- When you install a Replication Server that is not the ID Server, copy the ID Server Name, ID Server User, and ID Server Password from the "ID server " section of the worksheet.

Task

1. From the New Replication Server window, select ID Server Information.
   - If the Replication Server you are installing is not the ID Server, you see the ID Server Name window. Select ID Server Name and type the name of the ID Server for the replication system from the worksheet you completed for the ID Server. Then press Ctrl+A to accept your changes.
   - If the Replication Server you are configuring is the ID Server, the ID Server Information window appears.

   The Starting Replication Server ID and Starting Database ID appear only when you configure the ID Server.

2. Complete the ID Server Information window with the information from the "ID Server" section of your worksheet.

If you do not make entries for the Starting Replication Server ID and starting database ID, the default ranges are used, as shown in "Default ID Number Ranges and ID Number Allocation for ID Server Domains" section.

3. Press Ctrl+A to accept your changes and return to the New Replication Server windows.

See also
- *ID Server* on page 7
- *Default ID Number Ranges and ID Number Allocation for ID Server Domains* on page 19

Selecting the RSSD Type

Determine the RSSD type that you want for your replication system.

Before you can enter your Replication Server System Database (RSSD) information, you must determine if you want an RSSD or an embedded Replication Server System Database (ERSSD).

In the Replication Server System Database Choice window, select:

- **Yes** – if you want an ERSSD using SQL Anywhere. The text on the second line changes to:
  ```
  2. Embedded Replication Server System
  Database INCOMPLETE
  ```
  Select the **Embedded Replication Server System Database** option, to go to the ERSSD Name window, and then follow the steps to enter information for your ERSSD.
- **No** – if you do not want an ERSSD. This is the default value, and assumes you are creating a regular Replication Server System Database with Adaptive Server Enterprise. The second line on the window appears as:
  ```
  2. Replication Server Database on ASE INCOMPLETE
  ```
  Select the Replication Server System Database on ASE option, to go to the Replication Server System Database window, and then follow the steps to enter information for your RSSD.

See also
- *Entering Embedded Replication Server System Database Information* on page 48
- *Entering Replication Server System Database Information* on page 49

Entering Embedded Replication Server System Database Information

Complete the ERSSD information dialog box.

1. If you selected an ERRSD in "Selecting RSSD type", enter the name of your SQL Anywhere server in the ERSSD window, and press Ctrl+A to accept the name. The Embedded Replication Server System Database window appears.

Note: If you are configuring a Replication Server that previously existed, you must delete the previous ERSSD database (`*.db`), the previous transaction log file (`*.log`), and the previous transaction log mirror file (`*.mlg`) before you start **rs_init**.

2. You see the following options, each filled with a default value:

   - ERSSD Database Directory
   - ERSSD Transaction Log Directory
   - ERSSD Backup Directory
   - ERSSD Error Log Directory
   - ERSSD Interface Information

   Select each item to change the default value to the value you recorded in the Replication Server installation worksheet. See *Replication Server Administration Guide Volume 1 > Manage a Replication System* for more information on value requirements for ERSSDs.

3. Press Ctrl+A to accept the values and return to the Replication Server System Database Choice window.

4. Press Ctrl+A to return to the New Replication Server window.

Entering Replication Server System Database Information

Complete the Replication Server System Database Information dialog box.

1. In the New Replication Server window, select Replication Server System Database.

   Note: If you are configuring a Replication Server that previously existed, you must delete the previous RSSD before you start **rs_init**.

2. Select each item on the window and enter the value recorded on your worksheet. If you use a default value, copy it from the dialog box to your worksheet.

   For "RSSD will be replicated," select Yes if you will have more than one Replication Server in your environment.

3. Press Ctrl+A to accept the values. You see the New Replication Server window.

Entering RSSD Device Information

Complete the RSSD device information dialog box.

If you selected Yes for the Create RSSD option in the Replication Server System Database window, the "RSSD Device Information" option appears in the New Replication Server window.

Warning! If you selected No for Create RSSD in the Replication Server System Database window, you must already have created the RSSD. Skip this section and go to "Entering disk partition information".

1. From the New Replication Server window, select RSSD Device Information, then choose **Continue**.

    2. Complete the window using information from your worksheet. If you use a default value, copy it from the window to your worksheet.

    3. Press Ctrl+A to accept your entries and return to the New Replication Server window.

Entering RSSD RepAgent Information

Complete the RSSD RepAgent information dialog box.

If you selected Yes for the "RSSD will be replicated" option in the Replication Server System database window, the "Database Replication Agent" option appears in the New Replication Server window.

1. Select Database Replication Agent. Enter the RSSD Replication Server user name, and the Replication Server password from your worksheet.

> **Note:** Where necessary, **rs_init** uses the settings established for the Replication Server when configuring an RSSD RepAgent.

2. Press Ctrl+A to accept this information.

Entering Disk Partition Information

Complete the disk partition information dialog box.

Prerequisites
If the Replication Server disk partition is a raw device, the disk partition must exist before you can complete the Disk Partition Information window. If the partition does not exist, create it now. However, if the disk partition is an operating system file, Replication Server automatically creates the disk partition if the partition does already not exist.

Task

1. From the New Replication Server window, select Disk Partition.

2. Complete the window using the information you recorded on your worksheet.

You can add more partitions after Replication Server is installed. See the *Replication Server Design Guide* for help in determining how much space you need.

3. Press Ctrl+A to accept the values and return to the New Replication Server window.

See also
• *Creating a Disk Partition* on page 25

Entering Remote Site Connections Information

Modify the default values in the Remote Site Connections window, which **rs_init** provided based on the Replication Server name you entered.

1. From the New Replication Server window, select **Remote Site Connections**.
2. Complete the window using the information on your worksheet. If you use a default value, copy it to the "Remote site connections" section of the worksheet.
3. Press Ctrl+A to accept your changes and return to the New Replication Server screen.

Adding a Database to the Replication System

Add a database to the replication system. Each primary or replicate database in a replication system is managed by a Replication Server.

Prerequisites
Complete the Database Setup Worksheet.

Task

1. From the RS_INIT menu, select **Configure a Server Product**.
2. Select **Replication Server**. You see the Configure Replication System window.
3. Select **Add a Database to the Replication System**.
   rs_init indicates the status of each part of the installation with "Incomplete" or "Complete."
4. Select **Replication Server Information**.
5. Select **Replication Server Name**. Enter the name of the Replication Server from your worksheet. Press **Ctrl**+**A** to accept your changes.
6. Complete the window, using the information you recorded on your worksheet.

   If the status of Replication Server Interfaces Information is "Incomplete," add an entry to the interfaces file for the Replication Server.
7. Press **Ctrl**+**A** to accept your changes and return to the Add Database to Replication System window.

See also
- *Database Setup Worksheet Sample* on page 26
- *Editing the Interfaces File* on page 47

Completing Database Information
Finalize the database information.

1. In the Add Database to Replication System window, select Database Information, then click **Continue**.

---

2. Complete the window using the information you recorded on the worksheet. If you use a default value, copy it from the window to your worksheet. The label refers to Adaptive Server as "SQL Server."

    Note: The password you specify in this step must be valid to log in to Adaptive Server as "sa". You cannot change the password in **rs_init** by specifying a different password in this window. To change the Adaptive Server "sa" password, log in to the Adaptive Server after installation and change the password using the alter user command.

3. Choose Yes for "Will the Database Be Replicated?" if:

    *   It contains primary data.
    *   It is part of a warm standby application.
    *   Applications execute replicated stored procedures in it.

4. If you are adding the active or standby database for a warm standby application:

    a) Choose Yes for "Is This a Physical Connection for Existing Logical Connection?"
    b) Specify the additional information in the Logical DB Setup window.

5. Press Ctrl+A to accept your changes, then go to "Completing the Configuration" section.

Completing Logical Connection Information

Finalize the information you entered in the Logical Connection information window.

Prerequisites

*   Set up a warm standby application between Adaptive Server databases. See *Replication Server Administration Guide Volume 2 > Manage Warm Standby Applications*.
*   Set up a warm standy application between Oracle databases. See *Replication Server Heterogeneous Replication Guide > Heterogeneous Warm Standby for Oracle*.

    Note: Before you add a database for a warm standby application, you must create the logical connection in the Replication Server.

Task

1. If you selected Yes to "Is This a Physical Connection for Existing Logical Connection?" in the Database Information window, Logical Database Setup selections appear. You must specify logical connection parameters.

    In the Database Information window, select Logical DB Setup.

    The Logical Connection Information window appears. When you choose "active" for the first item, only these options appear:
    *   Logical DS Name
    *   Logical DB Name

2. Complete the Logical Connection Information window, using the information you recorded on your worksheet. If you use a default value, copy it from the window to your worksheet.

3. Press Ctrl+A to accept your changes, and **rs_init** returns you to the Database Information window.

Completing the Configuration

Complete the replication system configuration.

1. If the Database Information window indicates that there are incomplete tasks on the Add Database to Replication System window, select each one and complete the required information.
   When all tasks are "Complete," press Ctrl+A. A message is displayed asking you if you want to execute the configuration.

2. Enter "y" to continue with the configuration.
   Messages appear on the window as the configuration progresses; these messages require no action.

   Do not interrupt the configuration. Most configurations require several minutes (sometimes longer) to complete.

   When the configuration is complete, **rs_init** displays "Configuration completed successfully."

3. Click **OK** to return to the Configure Replication System window.

   Note: To add other databases to the replication system, go to "Adding a database to the replication system" section, and repeat the procedure for each database.

4. Press Ctrl+X to navigate through the screens and exit the program. **rs_init** displays the path to the log file for the session before you return to the command prompt.

You can find out more about the progress of the installation by looking at the current **rs_init** log file. **rs_init** log files are stored in the init/logs subdirectory of the $SYBASE_REP directory.

Upgrade or Downgrade Replication Server

Upgrade or downgrade Replication Server, or migrate to 64-bit platforms.

Warning! Do not modify the secondary truncation point of Replication Agent, truncate the transaction log, or increase the generation number while upgrading.

See also
*   *Required Replication Servers* on page 6

Requirements for Upgrading

There are several requirements for the site, route, and system versions, mixed-version environments, and different platforms, when you upgrade Replication Server.

Site, Route, and System Version
If a replication system domain has Replication Server 15.7.1 SP100 and later, then the system version and route versions in the replication system domain must be version 12.6 and later.

If the Replication Server version, the site version, and the route version are earlier than 12.6, you must upgrade Replication Server to version 12.6 or later, set site version to 12.6 or later, and upgrade routes to 12.6 or later, before you can install Replication Server 15.7.1 SP100.

Mixed-Version Environments
For Replication Server 15.7.1 SP100 and later , mixed-version environments containing versions earlier than 12.6 are not supported. You can upgrade directly to Replication Server 15.7.1 SP100 from versions 12.6 and later, including any ESDs for versions 12.6 and later.

Note: To create a database subscription from a primary Replication Server with version 15.7.1 SP100, the replicate Replication Server must be version 15.7.1 SP100 or later.

SQL Anywhere Version for ERSSD
ERSSD requires Sybase SQL Anywhere. SQL Anywhere version 12 included with Replication Server 15.7.1 SP100 is available on: 64-bit Linux, Solaris SPARC, Solaris x64, HP-UX Itanium, and IBM AIX platforms. The ERSSD and Sample Replication Server, and the `ASA12` folder are not available on 64-bit Linux on POWER (IBM pSeries).

See also
*   *Commit a Local Site to a New Version Level* on page 67
*   *Replication Server Site Version* on page 68
*   *Replication Server Route Version* on page 69

Preparing to Upgrade

Prepare the Replication Server for upgrade.

1. Back up the RSSD or ERSSD. The upgrade process makes changes to the RSSD or ERSSD that you cannot roll back if the upgrade fails. Restore the RSSD or ERSSD from the backup if the upgrade fails.

2. Determine where you want to install Replication Server.

    If you plan to install the Replication Server software in a directory that:
    • Does not contain your existing Replication Server installation – verify that the `interfaces` file is accessible and see *Upgrading Replication Server* on page 57
    • Contains your existing Replication Server installation – see *Upgrades and Downgrades Using Existing Directories* on page 83

3. Read the release bulletin for your platform for any special installation instructions about mixed-version environments, installing, or upgrading.

4. Use **rs_helpsub** and **rs_helpdbsub** to check that all subscriptions are valid. The upgrade fails if there are subscriptions that are not in a valid state before upgrading.

5. Install Replication Server, following the instructions in the *Replication Server Installation Guide* for your platform.

6. Verify the system-wide and route versions and upgrade to version 12.6 if necessary:

    a) Execute **sysadmin system_version** on the current Replication Server. If the system version is 1260 or later, you can upgrade your Replication Server now. If the system version is earlier than 12.6, proceed to step 6b.
    ```
    sysadmin system_version
    go
    ```

    An example of the output is:
    ```
    The current system version is 1102.
    ```

    b) Set the system version on the ID Server to 1260:
    ```
    sysadmin system_version,1260
    go
    ```

    If setting the system version is unsuccessful, this message appears and you must proceed to step 6c:
    ```
    The system version could not be set. The specified
    version '1260' is higher than the lowest Replication
    Server version in the system '1250'
    ```

    If setting the system version is successful, this message appears and you can proceed to step 6d:

```
The system version has been reset from 1102 to 1260
```

c) Upgrade Replication Server to 12.6 and set the site version to 12.6 for any Replication Server in the domain that has a version earlier than 12.6. Repeat step 6b.

d) Execute **sysadmin system_version** again on the current Replication Server. If the system version is 12.6 or later, proceed to step 6e.

e) Execute **admin show_route_versions** to verify all route versions are 1260 or later. If the route version is lower than 12.6, see *Replication Server Administration Guide Volume 1 > Manage Routes > Upgrade Routes* for information in upgrading the route version. After you upgrade the routes, you can upgrade the current Replication Server.

Upgrading Replication Server

Upgrade Replication Server from version 12.6 and later.

1. *Stopping Replication Agent and Quiescing Replication Server*

Stop Replication Agent and quiesce Replication Server to purge all inbound queues.

2. *Deciding the Upgrade Method to Use*

Decide whether you want to upgrade Replication Server and associated databases using the simplified upgrade method with repserver or continue to upgrade with rs_init.

3. *Upgrades to RSSD or ERSSD and User Databases*

After you upgrade the Replication Server RSSD or ERSSD, you must apply upgrades to a primary, replicate, or standby database in the replication system to accommodate changes, such as installing new stored procedures, that support the new version of Replication Server to which you are upgrading, and features such as Multi-Path Replication™.

4. *Upgrading RSSD or ERSSD and User Databases with repserver*

Use the repserver utility to upgrade the Replication Server RSSD or ERSSD, and user databases in one step.

5. *Upgrading RSSD or ERSSD with rs_init*

(Optional) Use rs_init to upgrade the RSSDs or ERSSDs so they are compatible with the executable programs.

6. *Fixing a Failed Replication Server Upgrade*

Retry the upgrade process at this point if it fails.

7. *Fixing User Database Upgrades with sysadmin upgrade, "database"*

Apply upgrades to Adaptive Server or Sybase® IQ, primary, replicate, or standby databases in the replication system that failed to upgrade during the automatic upgrade process, or to databases that you excluded from the automatic upgrade process.

8. *Upgrading an Adaptive Server User Database with rs_init*

If you use rs_init to upgrade Replication Server and the RSSD or ERSSD, you must apply upgrades to a primary, replicate, or standby database in the replication system to

accommodate changes, such as installing new stored procedures, that support the new
version of Replication Server to which you are upgrading, and features such as Multi-Path
Replication™.

9. *Upgrading Non-ASE User Databases and User Databases Not Supported by sysadmin
upgrade*

Use the corresponding upgrade scripts for databases such as Oracle, IBM DB2 UDB, and
Microsoft SQL Server that are not supported by the sysadmin upgrade, database command
and rs_init.

10. *Commit a Local Site to a New Version Level*

After you upgrade a Replication Server and its user databases, you must decide whether to
commit to the new Replication Server version level. Decide whether to set the replication
site version to the new version and whether to upgrade the route version to the new version.

11. *Backing Up the RSSD or ERSSD*

Back up the RSSD or ERSSD after upgrading the Replication Server, setting its site
version, and performing route upgrades to the relevant routes.

Stopping Replication Agent and Quiescing Replication Server

Stop Replication Agent and quiesce Replication Server to purge all inbound queues.

Replication Server 15.7.1 SP100 is available only on 64-bit platforms for all UNIX and Linux
operating systems. Therefore, upgrading to version 15.7.1 SP100 may include upgrading
from a 32-bit platform which requires a purge of inbound queues. See the *Replication Server
Release Bulletin* for supported 64-bit platforms.

1. On Replication Server, stop Replication Agent for all primary databases and system
databases:
```
suspend log transfer from all
```

2. If you are using Adaptive Server as the RSSD, stop RepAgent for the RSSD:
```
sp_stop_rep_agent RSSD_name
```

3. If there are incoming routes to the Replication Server you are upgrading, log in to the
source Replication Server and suspend all routes to the Replication Server, *dest_rs*, you are
upgrading:
```
suspend route to dest_rs
```

4. Verify that the Replication Server queues have been drained out and the Replication Server
you are upgrading has been quiesced by executing this command on the Replication
Server:
```
admin quiesce_check
```

If Replication Server is not yet quiesced, retry with **admin quiesce_force_rsi**.

Warning! You may lose data if you proceed with the upgrade process and Replication
Server is not quiesced.

Deciding the Upgrade Method to Use

Decide whether you want to upgrade Replication Server and associated databases using the simplified upgrade method with **repserver** or continue to upgrade with **rs_init**.

1. Use the **repserver** utility to upgrade the Replication Server RSSD or ERSSD, and user databases in one step.

   Optionally, continue to use **rs_init** to upgrade the Replication Server RSSD or ERSSD, and user databases in several steps.

2. Upgrade user databases in the replication system only if they failed to upgrade during seamless upgrade or if you excluded them from the upgrade process.

   To upgrade the user databases, use one of:
   - *Fixing User Database Upgrades with sysadmin upgrade, "database"* on page 65
   - *Upgrading an Adaptive Server User Database with rs_init* on page 66

   You can use **sysadmin upgrade, "database"** or **rs_init** to upgrade the user databases.

Upgrades to RSSD or ERSSD and User Databases

After you upgrade the Replication Server RSSD or ERSSD, you must apply upgrades to a primary, replicate, or standby database in the replication system to accommodate changes, such as installing new stored procedures, that support the new version of Replication Server to which you are upgrading, and features such as Multi-Path Replication™.

Upgrading an RSSD or ERSSD may add new Replication Server system tables, add new rows or columns to existing tables, or install new stored procedures. To accommodate changes that support the new version of Replication Server to which you are upgrading, **repserver** or **rs_init** applies upgrades to user databases such as a primary, replicate, or standby database in the replication system.

If you do not apply the upgrades to replicate databases such as:

- Adaptive Server – you cannot use Multi-Path Replication for replication to Adaptive Server databases after you upgrade Replication Server to 15.7 and later.
- Sybase IQ – Replication Server suspends replicate connections to the Sybase IQ replicate databases after you upgrade Replication Server to 15.7.1 or later, and you see a "Awaiting Upgr" status if you use **admin who**. You can upgrade the Sybase IQ database using one of:
  - **repserver -upgr** – upgrades Replication Server at the same time
  - **sysadmin upgrade,'database'** – after upgrading Replication Server with **rs_init** or **repserver** with the **-nodb** option

Upgrading RSSD or ERSSD and User Databases with repserver

Use the **repserver** utility to upgrade the Replication Server RSSD or ERSSD, and user databases in one step.

1. If the Replication Server uses Adaptive Server for the RSSD, grant sa_role to the primary user at the RSSD:

    ```
    sp_role 'grant', sa_role, primary_user
    ```

    You can revoke the "sa" role after the upgrade process is finished, by executing:

    ```
    sp_role 'revoke', sa_role, primary_user
    ```

    > **Note:** The *RSSD_primary_user* in the `.cfg` file must have a replication role or "sa" role. The *RSSD_primary_user* may not necessarily be the sa user ID or the user that has been entered in the **rs_init** pane.

2. Grant sa_role to the maintenance user for each user database.
3. Shut down the preupgrade version of Replication Server.
    a) Use **isql** to log in to the Replication Server as the system administrator:

    ```
    isql -Usa -Psa_password - Srs_name
    go
    ```

    b) Enter:

    ```
    shutdown
    go
    ```

4. Log in as the sybase user and change to the directory where you installed the new Replication Server—$SYBASE.
5. Set SYBASE environment variables to the directory where Replication Server 15.7.1 SP100 is installed by using the **SYBASE.sh** or **SYBASE.csh** shell scripts.
6. Start the new version of Replication Server in upgrade mode. Enter:

    ```
    repserver -upgr -S rs_name -A  erssd_release_dir
     -C config_file -I interfaces_file -E errorlog_file
    ```

    In upgrade mode, Replication Server upgrades the ERSSD or RSSD it is using, automatically connects to each user database to which it has maintenance user access, and where a connection exists, applies the upgrade scripts to the database.

    The full syntax is:

    ```
    repserver [-C config_file] [-i id_server]
    [-S rs_name] [-I interfaces_file]
    [-E errorlog_file] [-M] [-v] [-K keytab_file]
    [-upgr] [-A erssd_release_dir] [-purgeq]
    [-nodb {all|dbid_1[,dbid_2[,dbid_3[,…]]]]}
    [-e]
    ```

    where:
    *   **-upgr** – instructs Replication Server to start the upgrade
    *   **-A** *erssd_release_directory* – specifies the location of the release directory of the ERSSD you are going to upgrade if Replication Server is using an ERSSD, for example:
        `/sybase/REP-15_5/ASA11`
        If you do not include the **-A** option, Replication Server obtains the release directory location from the Replication Server configuration file if the configuration file

contains the information. Replication Server ignores the release directory location in the configuration file if you specify the **-A** option because what you specify manually in the **repserver** command overrides configuration file settings.

- **-purgeq** – purges transactions from the inbound queue. You must use this option if you are upgrading from a version of Replication Server earlier than 15.5.
- **-nodb all** – excludes all user databases from the upgrade process
- **-nodb** *dbid_1*[,*dbid_2*[,*dbid_3*[,...]]] – excludes specific databases from the upgrade process. Separate multiple database IDs with a comma and do not include a space between the IDs. For example:

```
repserver –upgr . . .  -A . . .  -nodb 101,102,105
```

- **-e** – records the SQL statements that Replication Server sends to data servers when you enter the **-upgr** parameter for upgrading. Without the **-e** option, there is no record of the SQL statements generated. With or without the **-e** option, the upgrade process uses the Replication Server error log file to record any errors that occur during the upgrade process, and why any user database was not upgraded.

You can use only the **-A**, **-purgeq**, **-nodb**, and **-e** options when you use **-upgr**. See *Replication Server Reference Manual > Executable Programs > repserver* for descriptions of the other **repserver** parameters.

For example, if you are upgrading the NY_RS Replication Server from version 15.5 to 15.7.1 SP100, and before the upgrade, NY_RS is using the /sybase/REP-15_5/ ASA11 ERSSD release directory location, the *ny_rs.cfg* configuration file, the interfaces file, and the ny_rs_errorlog error log file, enter:

```
repserver -upgr –SNY_RS –A/sybase/REP-15_5/ASA11 –Cny_rs.cfg –
Iinterfaces –E ny_rs_errorlog
```

If the upgrade is successful, the upgrade mode shuts down Replication Server and you do not see any warning or error messages.

7. Restart Replication Server using the appropriate executables for the new version.

Warning! Some Replication Server configuration options with newer versions may have a wider valid range. If you increase any of these configuration options after upgrading, you must reinstate the old values before downgrading. Otherwise, the Replication Server may not start or may behave incorrectly.

---

8. If you installed the new Replication Server software in a different directory than the earlier version, modify the runserver files for the Replication Server to replace the old binary with new binary in the new version directory.

9. If you are using Adaptive Server as the RSSD, restart the RepAgent for the RSSD:

```
sp_start_rep_agent RSSD_name
```

10. Resume log transfer to Replication Server from all Replication Agents for primary databases, and RSSD or ERSSD:

```
resume log transfer from all
```

11. If there are incoming routes to *dest_rs*, the Replication Server you upgraded and now restarted, log in to the source Replication Server and resume all routes to *dest_rs* to allow

---

the source Replication Server to begin sending queued messages to *dest_rs*. At the source Replication Server, execute:

```
resume route to dest_rs
```

12. Check the upgraded Replication Server:

    a) Verify that the RSSD or ERSSD version of the upgraded Replication Server is correct. You can check one of:

        • Replication Server error log – you see:

          ```
          Upgraded RSSD successfully. The current RSSD version is
          '1571100'.
          ```

        • `rs_config table` – log in to **isql** and enter:

          ```
          select * from rs_config where
          optionname='current_rssd_version'
          ```

    b) Check that the Replication Server modules are running by executing **admin who** with the relevant options.

13. If you have more Replication Servers to upgrade, repeat steps 1 to 12.

Upgrading RSSD or ERSSD with rs_init

(Optional) Use **rs_init** to upgrade the RSSDs or ERSSDs so they are compatible with the executable programs.

Upgrading an RSSD or ERSSD may add new Replication Server system tables, add new rows or columns to existing tables, or install new stored procedures. Upgrade an RSSD based on the information from the Replication Server installation worksheet for the Replication Server you are upgrading. You must separately use **rs_init** to upgrade any primary, replicate, or standby user databases in the replication system to accommodate changes, such as installing new stored procedures, that support the new version of Replication Server to which you are upgrading.

1. Log in as the "sybase" user and change to the directory where you installed Replication Server—$SYBASE.

2. Set SYBASE environment variables to the directory where Replication Server 15.7 is installed by using the **SYBASE.sh** or **SYBASE.csh** shell scripts.

3. Verify that the RSSD and Replication Servers that you are upgrading are running.

4. If the Replication Server uses Adaptive Server for the RSSD, grant sa_role to the primary user at the RSSD:

   ```
   sp_role 'grant', sa_role, primary_user
   ```

   You can revoke the sa_role after the upgrade process is finished, by executing:

   ```
   sp_role 'revoke', sa_role, primary_user
   ```

   Note: The *RSSD_primary_user* in the `.cfg` file must have a replication role or "sa" role. The *RSSD_primary_user* may not necessarily be the sa user ID or the user that has been entered in the **rs_init** pane.

5. If the Replication Server uses an ERSSD, edit the `rs_name.cfg` file, where *rs_name* is your Replication Server:

   a) Add the **erssd_release_dir** parameter to specify the location of the SQL Anywhere ERSSD database that you are going to upgrade. For example:

   ```
   erssd_release_dir=/sybase/REP-15_5/ASA11
   ```

   b) Add the **erssd_start_cmd** parameter to specify the command to start the previous version of ERSSD.

   You can find an example of the parameter in the Replication Server error log by searching for "ERSSD start command". At the entry for "ERSSD start command", you see for example:

   ```
   ERSSD start command: /sybase/REP-15_5/ASA11/bin/dbspawn
   -f -q /sybase/REP-15_5/ASA11/bin/dbsrv11
   -s none -ti 0 -x "tcpip(PORT=15501;DOBROAD=NO;BLISTENER=NO)"
   -o /sybase/REP-15_5/errorlog/rs155_prs_ERSSD.out
   /sybase/REP-15_5/dbfile/rs155_prs_ERSSD.db
   ```

   c) From the `interfaces` file, obtain the name and location of the Replication Server you are upgrading, and the port number of the Replication Server RSSD.

6. If the Replication Server uses an ERSSD, and is performing routing, edit the `rs_name.cfg` file, where *rs_name* is your Replication Server:

   a) Add the **erssd_ra_release_dir** parameter to specify the location of the SQL Anywhere ERSSD database that you are going to upgrade. For example:

   ```
   erssd_ra_release_dir=/sybase/REP-15_5/ASA11
   ```

   b) Add the **erssd_ra_start_cmd** parameter to specify the command to start the previous version of ERSSD.

   Obtain the name of the ERSSD Replication Agent and the location of the `interfaces` file from the Replication Server log file by searching for "Starting embedded RSSD Replication Agent" and using the same command format from the next line: "Executing command." For example:

   ```
   erssd_ra_start_cmd=/sybase/REP-15_5/ASA11/bin/dbltm
   -ud -S SAMP_RS_ERSSD_ra
   -C /sybase/REP-15_5/samp_repserver/SAMP_RS_ERSSD_ra.cfg
   -ot /sybase/REP-15_5/samp_repserver/SAMP_RS_ERSSD_ra.out
   -I /sybase/interfaces
   ```

7. If you are upgrading from Replication Server version 12.6 to version 15.7.1 SP100 and if the Replication Server uses an ERSSD, include the `lib` subdirectory of the `ASA` directory in the LD_LIBRARY_PATH environment variable:

   ```
   LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/work/RS126/REP-12_6/ASA8/lib
   export LD_LIBRARY_PATH
   ```

8. Start **rs_init**:

   ```
   $SYBASE/$SYBASE_REP/install/rs_init
   ```

   You see the RS_INIT menu.

9. Select **Configure a Server Product**. Choose **Continue**.

10. Select **Replication Server**.

11. Select **Upgrade an Existing Replication Server**.

12. Complete the Upgrade Existing Replication Server window with the information from the "Replication Server installation worksheet" for the Replication Server you are upgrading.

    a) Enter the name of the Replication Server with the RSSD you are upgrading.

    b) Enter the login name for the Replication Server "sa" user. The default value is "sa."

    c) Enter the password for the "sa" user on the Replication Server.

    d) Enter the path name of the Replication Server configuration file.

    e) Press **Ctrl+A** to save your changes. You see:
```
Execute the Replication Server tasks now?
```

13. Enter "y" to continue the RSSD upgrade.

rs_init displays informational messages as it loads the upgrade script into the RSSD. When the upgrade is complete, you see:
```
RSSD successfully upgraded from old_rel_no to new_rel_no.
Replication Server 'rs_name' can now be restarted. Task to
upgrade the RSSD succeeded. Configuration completed
successfully.
```
where:

- *old_rel_no* – is the Replication Server version from which you are upgrading.
- *new_rel_no* – is the new Replication Server version.
- *rs_name* – is the name of the Replication Server.

14. If you installed the new Replication Server software in a different directory than the earlier version, modify the runserver files for the Replication Servers so that they use the programs in the new version directory.

15. Restart Replication Server using the appropriate executables for the new version.

Warning! Some Replication Server configuration options with newer versions may have a wider valid range. If you increase any of these configuration options after upgrading, you must reinstate the old values before downgrading. Otherwise, the Replication Server may not start or may behave incorrectly. The Replication Server upgrade process prints all the current configuration option values in **rs_config** to **rs_init** log, so that you can restore the former settings before downgrading.

---

    a) Modify the old `runserver` file, run_*servername*, where *servername* is your Replication Server name, to replace the old binary with the new binary in `$SYBASE/REP-15_5/bin/repserver` and start Replication Server from `$SYBASE/REP-15_5/bin/repserver`.

    b) If you are using Adaptive Server as the RSSD, restart the RepAgent for the RSSD:
```
sp_start_rep_agent RSSD_name
```

    c) Resume log transfer to Replication Server from all Replication Agents for primary databases, and RSSD or eRSSD:

```
resume log transfer from all
```

d) If there are incoming routes to *dest_rs*, the Replication Server you upgraded and now restarted, log in to the source Replication Server and resume all routes to *dest_rs* to allow the source Replication Server to begin sending queued messages to *dest_rs*. At the source Replication Server, execute:

```
resume route to dest_rs
```

16. If you have more Replication Servers to upgrade, repeat the upgrade procedures.

Fixing a Failed Replication Server Upgrade

Retry the upgrade process at this point if it fails.

1. Restart the old Replication Server against the RSSD and fix any problem that may have caused the error.

2. Re-run the upgrade process after you fix the error.

You can do this multiple times until the upgrade succeeds.

Note: You cannot start a new Replication Server against a partially upgraded RSSD if you used **rs_init**. However, you can start a new Replication Server in upgrade mode as many times as needed, using the **-upgr** option with **repserver**, against a partially upgraded RSSD to continue the upgrade process or to fix upgrade problems.

Fixing User Database Upgrades with sysadmin upgrade, "database"

Apply upgrades to Adaptive Server or Sybase® IQ, primary, replicate, or standby databases in the replication system that failed to upgrade during the automatic upgrade process, or to databases that you excluded from the automatic upgrade process.

The upgrades are required to accommodate changes, such as installing new stored procedures, that support the new version of Replication Server to which you are upgrading.

1. Enter **admin version, "connection"** at the upgraded Replication Server to identify the user databases that you must upgrade.

You see a list of the user databases and data servers, the database IDs, the corresponding Replication Server, and the status of the database. For example:

```
dbid    Name        Controller RS    Status
------  ----------  -------------    ------------------
101     pds.pdb01   rs_12            Database needs upgrade
102     pds.pdb02   rs_12            Database is not accessible
103     rds.rdb01   rs_12            Database has been upgraded
```

The "Not accessible" status means that Replication Server cannot connect to this user database because the database is unavailable, or because the maintenance user ID that Replication Server uses to connect to the database has insufficient privileges to connect.

2. Enter **sysadmin upgrade, "database"** at the upgraded Replication Server to upgrade the user databases served by the Replication Server:

```
sysadmin upgrade, "database" {,data_server, database | all}
```

where:

- *dataserver_name*, *database_name* – specifies a database to upgrade. You must enter a separate command for each database.
- **all** – upgrades all databases served by the Replication Server. Replication Server displays error messages if a database does not meet the conditions for upgrade.

For example, to upgrade the `pdb01` database in the `pds` data server, enter:

```
sysadmin upgrade, database, pds, pdb01
```

If any of the databases fail to upgrade, see the Replication Server error log for the reasons such as:

```
Database data_server.database is not accessible.
Fail to upgrade data_server.database.
```

Upgrading an Adaptive Server User Database with rs_init

If you use **rs_init** to upgrade Replication Server and the RSSD or ERSSD, you must apply upgrades to a primary, replicate, or standby database in the replication system to accommodate changes, such as installing new stored procedures, that support the new version of Replication Server to which you are upgrading, and features such as Multi-Path Replication™.

If you do not apply the upgrades to replicate databases such as:

- Adaptive Server – you cannot use Multi-Path Replication for replication to Adaptive Server databases after you upgrade Replication Server to 15.7 and later.
- Sybase IQ – Replication Server suspends replicate connections to the Sybase IQ replicates after you upgraded Replication Server to version 15.7.1 or later, and you see a "Awaiting Upgr" status if you use **admin who**.

1. Start **rs_init**.
2. Select **Upgrade an existing database in the replication system** from the Configure Replication System window, then choose **Continue**.
3. Enter the information from the "Database Setup Worksheet" for the upgraded Replication Server:
   - Name of the Adaptive Server that manages the database you are upgrading.
   - Name of the database you are upgrading.
   - Login name for the "sa" user on the Adaptive Server that manages the database that you are upgrading. The default value is "sa."
   - Adaptive Server "sa" user password.
   - Login name for the maintenance user on the Adaptive Server that manages the database that you are upgrading. The default is "database_maint," where "database" is the name of the database you are upgrading.
4. Select **Continue**. **rs_init** performs the upgrade.

5. If you have more Replication Servers or databases to upgrade, repeat the procedures to stop Replication Agent and quiesce Replication Server, upgrade RSSD or ERSSD, and upgrade Adaptive Server databases.

Upgrading Non-ASE User Databases and User Databases Not Supported by sysadmin upgrade

Use the corresponding upgrade scripts for databases such as Oracle, IBM DB2 UDB, and Microsoft SQL Server that are not supported by the **sysadmin upgrade, database** command and **rs_init**.

1. Use the maintenance user login ID to connect to the user database.

   See the Replication Server Options documentation to configure a connection to a non-ASE database.

2. Find the relevant script in the `scripts` subdirectory of your Replication Server installation directory.

   The scripts are named hds_*dataserver_make*_upgradedb.sql where *dataserver_make* is the replicate data server type.

3. Run the script to upgrade the user database.

   Modify or run the script unchanged depending on the way Replication Server connects to the user database. If the connection is through:

   - Enterprise Connect™ Data Access (ECDA) – run the script unchanged. For example, to upgrade an IBM DB2 UDB user database, use ECDA for ODBC with the **isql -i** option to run the **hds_db2_upgradedb.sql** script on the user database.
   - ExpressConnect – modify the script to allow the tool of your choice to run it. For example, to upgrade an Oracle database, remove the **go** commands in the **hds_oracle_upgradedb.sql** script and load the script manually .

Commit a Local Site to a New Version Level

After you upgrade a Replication Server and its user databases, you must decide whether to commit to the new Replication Server version level. Decide whether to set the replication site version to the new version and whether to upgrade the route version to the new version.

The system version is the version of the replication system. Set the system version at the ID Server with **sysadmin system_version**.

In a replication system that includes Replication Server versions 12.6 through 15.7.1 SP100, and for which the site version is set to its software release level and the system version is 1260 or higher, each Replication Server can use all the features of its release. Such a system is called a mixed-version system.

However, in a mixed-version system, interaction between Replication Servers of different version levels is limited, and information associated with new features may not be available to Replication Servers of earlier versions. Replication Servers of the same version can exchange full information about the software features they support.

If, for example, you are upgrading to version 15.7.1 SP100, you can use new Replication Server features that interact with other Replication Servers only if all these conditions are true:

- The system version has been set to 1260 or higher.
- The site version for both Replication Servers has been set to the current site version, such as 1571100.
- The routes between the Replication Servers have been upgraded.

See *Replication Server Reference Manual > Replication Server Commands > **sysadmin system_version***.

See also
- *Preparing to Upgrade* on page 56
- *Backing Up the RSSD or ERSSD* on page 72

System Version and Software Version Relationship

You can use the full capabilities of software version 15.5 and later when the system version is at 1260 or later.

Replication Server Software Version	System Version 1260 and Later
15.7.1 SP100	Full capability when site version is set to 1571100
15.7.1	Full capability when site version is set to 1571
15.7	Full capability when site version is set to 1570
15.6 and 15.5	Full capability when site version is set to 1550
15.2	Full capability when site version is set to 1520
15.1	Full capability when site version is set to 1510
15.0 and 15.0.1	Full capability when site version is set to 1500
12.6	Full capability when site version is set to 1260

For example, new features of version 15.7.1 SP100 are available only when the system version is at least 1260 and the site version is set to 1571100.

If a software version is supported with full capability at a given system version, you can install new Replication Servers of that software version.

Replication Server Site Version

The site version determines which Replication Server features you can use.

If, for example, you have Replication Servers with a site version set to version 15.0, you can create replication definitions for tables containing the `bigint` datatype. Such replication

definitions are distributed to other Replication Servers with site versions that are at the same version level or higher. Replication Servers of earlier versions do not receive information about these replication definitions.

After you upgrade Replication Server and its user databases, and set the system version to the new version, you must set the Replication Server site version to the new software version level before you can use the new Replication Server features.

You cannot downgrade after you set the site version. You must set the site version for all Replication Servers that require it, and upgrade the routes. Information needed for the new features cannot be propagated to other sites until the corresponding routes are upgraded.

See also
*   *Replication Server Route Version* on page 69

Setting the Replication Server Site Version to the New Version
Set the Replication Server site version to the new version to use the new features of Replication Server.

1. Log in to the current Replication Server and execute **sysadmin site_version,** *new_site_version* to complete the upgrade.
    For example, to set the site version to 1570:

    ```
    sysadmin site_version, 1570
    ```

    The *Replication Server Release Bulletin* may instruct you to set the site version to a more recent version level.

    See *Replication Server Reference Manual > Replication Server Commands >* **sysadmin site_version**.
2. If the Replication Server uses an ERSSD, remove the **erssd_release_dir** and **erssd_start_cmd** parameters, and all options and values you entered earlier for these parameters from the *rs_name*.cfg file of the new version of Replication Server.

Replication Server Route Version
The route version is the earlier of the two site version numbers of the source and destination Replication Servers of the route. After you upgrade the source and destination Replication Servers on either end of a route and also set their site versions to a higher Replication Server version, you must upgrade the route.

Upgrading the route allows the Replication Servers to exchange information about newer software features. Upgrading a route rematerializes the data in the system tables in the RSSD and makes information associated with new features available to a newly upgraded Replication Server. After upgrading, new types of information that were not previously allowed can be exchanged.

Replication Server version 11.5 and later uses route version information to determine which feature set you can use with the route and which data to propagate to other sites. If the route

version is lower than the earliest site version, you must perform a route upgrade. For example, information necessary for version 15.7.1 SP100 features is not propagated to other sites until the corresponding route versions are upgraded to 1571100.

After you upgrade to Replication Server 15.7.1 SP100 and set the site version to 1571100, you must upgrade each route for which this Replication Server is a source or destination, and the site versions of the source and destination are at 1571100 or later. Set the new site version and upgrade the routes only to use new features across multiple Replication Servers that support the new feautures.

How you upgrade routes depends on how you use Replication Server, and whether you are using a mixed-version system.

If the source Replication Server is version 15.7 or later, you can use **sysadmin upgrade, "route"** to upgrade a route regardless of the destination Replication Server version. Otherwise, use the Replication Manager plug-in of Sybase Central™.

Upgrading Routes
Set the route version to the new version.

1. Execute **admin version, "route"** to report the routes that you need to upgrade that start from and go to the Replication Server you are upgrading. An example of the report:

```
Source    Destination   Route     Proposed   Status
                         Version   Version


------   -----------   -------   --------   ------------------
NY_RS    LON_RS         1500      1571100    Need route upgrade
```

2. Execute **sysadmin upgrade, "route",** *dest_rs_name* on the Replication Server that originates the route you want to upgrade where *dest_rs_name* is the destination Replication Server of the route you are upgrading.

   The user ID and password you use to execute the command must also exist at the destination Replication Server, and the RSSD of the destination Replication Server. This user ID must have sa permission at the destination Replication Server and dbo permission at the RSSD of the destination Replication Server.

   For example, to upgrade the route from NY_RS to LON_RS in the example replication system in step 1, execute:
```
sysadmin upgrade, "route", LON_RS
```

   When you execute the command, you see:
```
Route upgrade for route 'NY_RS.LON_RS' is in progress in the
background"
```

   In the background, NY_RS starts a routing thread that performs these procedures:

- NY_RS determines if the route upgrade requires RSSD materialization. If it does, NY_RS proceeds to the next step. Otherwise, NY_RS places a commit marker in the RSSD of LON_RS to alter the route version and completes the route upgrade.
- NY_RS places a begin marker in the RSSD of NY_RS.
- NY_RS checks every two seconds for the arrival of the begin marker at the RSSD of LON_RS. If the begin marker does not arrive at the RSSD of LON_RS within the defined threshold of one minute, NY_RS terminates the upgrade process.
- When the begin marker arrives at the RSSD of LON_RS, NY_RS instructs LON_RS to enter hibernation mode.
- The command instructs NY_RS to process the set of route upgrade scripts.
- NY_RS places a commit marker in the RSSD of LON_RS to alter the route version and then NY_RS completes the route upgrade.
- NY_RS instructs LON_RS to leave the hibernation mode.

3. Execute **admin version, "route"** to check the status of the route upgrade.

If:

- The route upgrade succeeds, the route should not be listed in the output.
- The route upgrade fails and you need to recover the route from the upgrade, you see something similar to:

```
Source   Destination  Route    Proposed  Status
                       Version  Version
------   -----------  -------  --------  ----------------
NY_RS    LON_RS       1500     1571100      need route
                                          upgrade recovery
```

- If the route upgrade did not proceed and if there are still routes that you need to upgrade, you see something similar to:

```
Source   Destination  Route    Proposed  Status
                       Version  Version
------   -----------  -------  --------  ----------------
NY_RS    ROM_RS       1500     1571100      need route
                                          upgrade
```

You can also check the error log of *source_rs_name*, the Replication Server where the route starts, for the status of the route upgrade. In a:

- Successful route upgrade that includes materialization to make information associated with new features available to a newly upgraded Replication Server:

```
Upgrade for route 'source_rs_name.dest_rs_name' is
complete.
```

- Successful route upgrade that does not need rematerialization:

```
Upgrade for route 'source_rs_name.dest_rs_name' is
complete. no copying of RSSD information is necessary.
```

- Failed route upgrade:

```
Route upgrade/upgrade recovery fails due to error. The
upgrade procedure is not completed. This routing thread
will in turn exit.
```

4. To recover the route upgrade if the route upgrade fails:

a) Execute **sysadmin upgrade,"route",** *dest_rs*, **"recovery"** at the source Replication Server.
For example, at NY_RS, execute:

```
sysadmin upgrade, "route", LON_RS, "recovery"
```

b) Execute **admin version,"route"** at NY_RS to verify that the route upgrade is completed.

5. Restart the primary Replication Server distributor thread connection from the primary database after you upgrade the route to support the replication of features included with the new version of Replication Server.
For example, to restart the thread from the `pdb` primary database in the PDS primary data server, enter:

```
suspend distributor PDS.pdb
go
resume distributor PDS.pdb
go
```

Backing Up the RSSD or ERSSD

Back up the RSSD or ERSSD after upgrading the Replication Server, setting its site version, and performing route upgrades to the relevant routes.

1. Use the Adaptive Server commands **dump database** or **dump transaction** to back up the RSSD, and the Replication Server **sysadmin erssd, backup** command to back up the ERSSD.

2. (Optional) You can restore the RSSD with the Adaptive Server commands **load database** or **load transaction**, or restore the ERSSD using the ERSSD recovery procedure.

See *Replication Server Administration Guide Volume 1 > Manage a Replication System > > Manage the Embedded Replication Server System Database* for information on recovery procedures.

3. (Optional) If you have a current backup, you can avoid potential problems that can occur if you load an RSSD that corresponds to an earlier version level.
After you load the backup, all queued messages from other Replication Servers can be recognized by this Replication Server and RSSD.

If you restore the RSSD from a backup performed before you set the site version, Replication Server may receive messages that the RSSD cannot accept.

See also
• *Commit a Local Site to a New Version Level* on page 67

Upgrading Adaptive Server in a Replication System

You can separately upgrade Adaptive Server in your replication system.

Prerequisites
Sybase strongly recommends you perform a dump database and dump transaction before upgrading Adaptive Server.

Task

1. *Suspend Replication and Transaction Activity in the Database*

   Suspend replication and transaction activity in the database. Replication activity includes creating and dropping both routes and subscriptions.

2. *Drain the Transaction Logs of the Primary Databases*

   Ensure that the Replication Server completely processes the preupgrade log for each primary database you are upgrading.

3. *Drain the RSSD Transaction Log*

   Create a replication definition to manually drain the RSSD transaction log. This ensures that Replication Server processes all transactions in the RSSD log before you upgrade databases if Replication Server has routes to other Replication Servers.

4. *Disable the Secondary Transaction Point*

   Turn off the secondary truncation point for the duration of the upgrade and when you upgrade a primary database, the Replication Agent cannot be running.

5. *Upgrade Adaptive Server*

   See the Adaptive Server Enterprise Installation Guide for upgrade instructions.

6. *Update the Adaptive Server System Catalog*

   To ensure that replication proceeds correctly for objects with quoted identifiers created in versions of Adaptive Server earlier than 15.7, update the Adaptive Server system catalog after you upgrade Adaptive Server, but before you restore replication.

7. *Restore Replication*

   Restore replication after you perform the upgrade procedure.

Suspending Replication and Transaction Activity in the Database

Suspend replication and transaction activity in the database. Replication activity includes creating and dropping both routes and subscriptions.

1. Verify that the subscriptions you have created with primary data in the databases being upgraded, have reached a "valid" state at the primary Replication Server.

   Do not upgrade while the subscriptions are being created.

Make sure no users create subscriptions for the data in the database you are upgrading until the upgrade procedure is finished.

2. Run **rs_helproute** in each RSSD being upgraded to determine its status.

    The status of all routes should be "Active." See *Replication Server Administration Guide Volume 1 > Managing Routes* to resolve route problems.

3. Shut down the applications that are using the databases you are upgrading.

4. Use the **admin who** command in Replication Server to identify the existing Data Server Interface (DSI) connections to the data server being upgraded.

5. Suspend all DSI connections to databases you are upgrading. For each database, issue:
    ```
    suspend connection to dataserver.database
    ```

Draining Transaction Logs for Primary Databases

Ensure that the Replication Server completely processes the preupgrade log for each primary database you are upgrading.

1. Wait for all remaining transactions to be replicated.

2. Execute:
    ```
    admin who, sqm
    ```

    Find the entry that corresponds to the inbound queue for this database by looking in the Info field for the *queue_number* and *queue_type* entry. For an inbound queue, the queue type is 1. Note the last segment:block entry for the queue.

3. Open the queue dump file:
    ```
    sysadmin dump_file, "file_name"
    ```

    where *file_name* is the file to which you are dumping.

4. Create a dummy table to check that the Replication Server has received the latest log record written in the log. You can drop this table later.
    ```
    create table dummy (c1 int, c2 char(255))
    go
    sp_setreptable dummy, true
    go
    begin tran
    go
    insert dummy values (1,'hello')
    go 10
    commit tran
    go
    ```

5. In the primary Replication Server, execute the **admin who, sqm** command until the last segment:block entry for the inbound queue changes.

6. In Replication Server, dump the last block of the inbound queue to the dump file you created in step 3:
    ```
    sysadmin dump_queue, queue_number, queue_type,
    last_seg, block, 1
    ```

Use the *queue_number*, *queue_type*, *last_seg*, and *block* values found in the output of the **admin who, sqm** command in step 5.

7. Use a text editor to examine the dump file to make sure it contains the transaction corresponding to the inserts you performed in step 4.

8. Repeat steps 5 through 7 until the transaction corresponding to the update is in the dump file. After draining the transaction logs, do not allow any other activity in the databases. If activity does occur, you must redrain the transaction logs.

Draining the RSSD Transaction Log

Create a replication definition to manually drain the RSSD transaction log. This ensures that Replication Server processes all transactions in the RSSD log before you upgrade databases if Replication Server has routes to other Replication Servers.

To make sure the transaction log is completely processed, create a replication definition in the primary Replication Server and verify that it appears in the replicate Replication Server RSSD. When the replication definition is in the replicate RSSD, the log is fully processed.

1. Log in to the primary Replication Server.

2. Create a temporary replication definition:

```
create replication definition rep_def_name
with primary at dataserver.database
with all tables named 'table_name'(column_name datatype)
primary key (column_name)
```

Provide the names for the data server, database, table, and column, and the datatype of the column. See the *Replication Server Reference Manual* for the complete syntax.

3. Log in to the replicate RSSD.

4. See whether the replication definition has arrived from the primary RSSD:

```
rs_helprep rep_def_name
```

When the replication definition has arrived in the replicate RSSD, the RSSD transaction log has been drained.

Disabling the Secondary Truncation Point

Turn off the secondary truncation point for the duration of the upgrade and when you upgrade a primary database, the Replication_Agent cannot be running.

1. Shut down the Replication Agents, or make sure that **dbcc logtransfer** is not running for the databases that are being upgraded.

2. Shut down Replication Servers for the RSSDs you are upgrading.

3. In each primary database including RSSDs, turn off the secondary truncation point:

```
use database
go
```

```
dbcc settrunc ("ltm", "ignore")
go
```

Repeat step 3 for each primary database and each primary RSSD.

Upgrading Adaptive Server

See the Adaptive Server Enterprise Installation Guide for upgrade instructions.

Updating the Adaptive Server System Catalog to Support Quoted Identifiers

To ensure that replication proceeds correctly for objects with quoted identifiers created in versions of Adaptive Server earlier than 15.7, update the Adaptive Server system catalog after you upgrade Adaptive Server, but before you restore replication.

1. Log in to **isql** at the primary Adaptive Server database.
2. Update the system catalog. Enter:

```
set nocount on
declare @cmd varchar(200)
        ,@cmdstat int
        ,@count int
        ,@dbid int
        ,@dbname varchar(30)
        ,@total int
select @cmdstat = config_admin(1, 102, 1, 1, NULL, NULL)
select @dbid = 1, @count = 0, @total = 0
while @dbid is not null
begin
  select @dbname = db_name(@dbid)
  select @cmd = 'select @count = count(1) from '+ @dbname
                + '.dbo.sysobjects'
                + 'where (sysstat2 & 134217728) = 0 '
                + 'and type = ''U'''
  exec (@cmd)
  if @count > 0
  begin
  print 'Object count in %1! is %2!', @dbname, @count
  select @total = @total + @count
  select @cmd = 'update ' + @dbname + '.dbo.sysobjects '
                + 'set sysstat2 = sysstat2 | 134217728 '
                + 'where (sysstat2 & 134217728) = 0 '
                + '  and type = ''U'''
  exec (@cmd)
  end
  select @dbid = min(dbid) from sysdatabases where dbid > @dbid
end
if @total > 0
  print 'You must restart ASE for changes to take effect.'
select @cmdstat = config_admin(1, 102, 0, 1, NULL, NULL)
```

Restoring Replication

Restore replication after you perform the upgrade procedure.

1. Zero out the locator in the RSSD for each replicated primary.

   Use **isql** to connect to the RSSD and execute:

   ```
   use RSSD
   go
   rs_zeroltm dataserver, database_name

   go
   ```

2. Set the secondary truncation point to "valid" in each primary database. Use **isql** to connect to the replicated primary database and execute:

   ```
   use database
   go
   dump tran database with truncate_only
   go
   dbcc settrunc("ltm","valid")
   go
   ```

Downgrading Replication Server

Revert to an earlier version of the software. You cannot downgrade an RSSD to a version below the site version.

Prerequisites

* Read the *Replication Server Release Bulletin* for any information about downgrading your software. In particular, find out if an RSSD downgrade is required to return to the earlier version. When you downgrade to an earlier software version, you must install the earlier software. Read the release bulletin for the earlier version of Replication Server. Then refer to the installation or configuration guide for that version for installation instructions.

  Note: If an RSSD downgrade is not required, you do not have to perform these steps. Reinstall the earlier software version following the instructions in *Replication Server Installation Guide*.

* Back up your system.

  If you are downgrading to a directory other than the existing Sybase version directory, verify that the appropriate interfaces file is accessible.

* Back up the RSSD for the Replication Server you plan to downgrade.

  Warning! Some Replication Server configuration options with newer versions may have a wider valid range. If you increase any of these configuration options after upgrading, you must reinstate the old values before downgrading. Otherwise, the Replication Server may not start or may behave incorrectly. The Replication Server upgrade process prints all the

current configuration option values in **rs_config** to **rs_init** log, so that you can restore the former settings before downgrading.

Task

1. Stop Replication Agent for all primary databases and system databases by executing this command on Replication Server:
```
suspend log transfer from all
```
2. If you are using Adaptive Server as your RSSD, stop RepAgent for the RSSD:
```
sp_stop_rep_agent RSSD_name
```
3. If you are going to downgrade a replicate Replication Server, log in to the primary Replication Server and suspend all the routes to the replicate Replication Server:
```
suspend route to replicate_Replication_Server
```
4. Verify that the Replication Server queues have been drained and that Replication Server has been quiesced:
```
admin quiesce_check
```

   Retry with **admin quiesce_force_rsi** if Replication Server is not yet quiesced.

   Warning! You may lose data if you proceed with the downgrade process without quiescing Replication Server.

5. Log in as the "sybase" user, and change to the $SYBASE_REP directory.
6. Verify that the RSSDs and the Replication Servers you plan to downgrade are running.

   Note: The downgrade process purges all the Replication Server inbound queues.

7. Start **rs_init**:
```
$SYBASE/$SYBASE_REP/install/rs_init
```
   The RS_INIT menu appears.

8. Select **Configure a Server Product**.
9. Select **Replication Server**.
10. Select **Downgrade RSSD for an Existing Replication Server**.
11. Enter information from the "Replication Server Installation Worksheet" section for the Replication Server you are downgrading:
    • Name of the Replication Server for which you are downgrading the RSSD.
    • Login name for the Replication Server "sa" user. The default value is "sa."
    • Password for the Replication Server "sa" user.
    • Path name of the Replication Server configuration file.
12. Press **Ctrl+A** to save your changes. You see:
```
Execute the Replication Server tasks now?
```

13. Enter "y" to continue.

Informational messages appear as **rs_init** loads the downgrade script into the RSSD. When the downgrade is complete, you see these messages:

```
RSSD successfully downgraded to rel_no. Replication Server
'rs_name' can now be restarted. Task to downgrade the RSSD
succeeded. Configuration completed successfully.
```

where:

- *rel_no* – is the Replication Server version from which you are downgrading.
- *rs_name* – is the name of your Replication Server.

Press Return after each message. You see the Configure Replication System screen.

14. Press **Ctrl**+**C** and enter "y" to exit **rs_init**.

15. Shut down all Replication Servers and Replication Agents on this machine.

16. Restart the Replication Servers and Replication Agents.

17. If you are using Adaptive Server as the RSSD, restart the RepAgent for the RSSD:

```
sp_start_rep_agent RSSD_name
```

Downgrade Restrictions

After you set the replication site version to a new level, you cannot downgrade any Replication Servers in the replication system below that version level, or install a Replication Server below that version level.

For example, if you upgrade to version 15.7.1 SP100, Replication Server may write some version 15.7.1 SP100 commands to the inbound queue after the upgrade. Versions of Replication Server earlier than 15.7.1 SP100 cannot recognize the version 15.7.1 SP100 commands.

To return to an earlier software version, reinstall all Replication Servers and re-create your replication applications.

If you do not want to use new features that depend on the site version, do not set the Replication Server site version after the upgrade. You can then downgrade to an earlier version if needed.

If you plan to downgrade in a directory other than the existing Sybase version directory, verify that the `interfaces` file is accessible.

If necessary, downgrade the RSSD for each Replication Server you are downgrading. You must downgrade the RSSD using the predowngrade— that is, the most recent—version of **rs_init**.

If the Replication Server uses an ERSSD, verify that the Replication Server configuration file contains the **erssd_release_dir** and **erssd_start_cmd** parameters. Ensure the values for these parameters should be the same as the values before upgrading.

Migrating Between 32-bit and 64-bit Platforms

Replication Server is available only on 64-bit platforms for all supported UNIX and Linux operating systems.

Prerequisites

* Back up your replication system. The upgrade process makes changes to the system databases that you cannot roll back if the upgrade fails. Restore the system databases if the upgrade fails.
* If you plan to install the Replication Server software in a directory other than an existing Sybase version directory, verify that the `interfaces` file is accessible.

Task

Note: If you have upgraded to a Replication Server version available only on 64-bit platforms, you cannot migrate back to a 32- bit platform. See the *Replication Server Release Bulletin* for supported 64-bit platforms.

1. Stop Replication Agent for all primary databases and system databases by executing this command on Replication Server:

   ```
   suspend log transfer from all
   ```
2. If you are using Adaptive Server as your RSSD, stop RepAgent for the RSSD:

   ```
   sp_stop_rep_agent RSSD_name
   ```
3. If you are going to migrate a Replication Server, log in to each upstream Replication Server and suspend all the routes to the Replication Server you are migrating:

   ```
   suspend route to replicate_Replication_Server
   ```
4. Verify that the Replication Server queues have been drained and that Replication Server has been quiesced:

   ```
   admin quiesce_check
   ```

   Retry with **admin quiesce_force_rsi** if Replication Server is not yet quiesced.

   Warning! You may lose data if you proceed with the downgrade process without quiescing Replication Server.
5. Stop Replication Server:
   a) Log in to the Replication Server as the system administrator:

   ```
   isql -Usa -Psa_password -Sservername
   ```
   b) Enter:

   ```
   1> shutdown
   2> go
   ```

6. If you are using an ERSSD, manually restart it:

a) Set the LD_LIBRARY_PATH environment variable:

```
$ export
LD_LIBRARY_PATH=$SYBASE/REP-15_5/ASA12/
lib:$SYBASE/REP-15_5/lib64:$SYBASE/REP-15_5/
lib:$SYBASE/REP-15_5/lib3p64/log4cxx/lib:$SYBASE/
OCS-15_0/lib:$SYBASE/OCS-15_0/lib3p64:$SYBASE/OCS-15_0/
lib3p
```

b) Start the ERSSD:

```
-o $SYBASE/REP-15_5/samp_repserver/SAMPLE_RS.log
$SYBASE/REP-15_5/samp_repserver/dbfile/SAMPLE_RS_ERSSD.db &
```

See your Replication Server log file for the actual commands to set LD_LIBRARY_PATH and to start the ERSSD by searching for "LD_LIBARY_PATH" and "ERSSD start command" respectively.

7. Load the migration script file into the ERSSD or RSSD. For example:

- ERSSD – load `$SYBASE/REP-15_5/scripts/rs_migration_asa.sql` into the ERSSD.
- RSSD – load `$SYBASE/REP-15_5/scripts/rs_migration_ase.sql` into the RSSD.

8. Start Replication Server from the new Replication Server executable.

See also

- *Start or Stop a Replication Server* on page 97

Upgrades and Downgrades Using Existing Directories

You can upgrade and downgrade Replication Serve if you install Replication Server into existing directories.

By default, the Replication Server installation process installs versions 15.5 to 15.7.1 SP100 in directories under the REP-15_5 directory. Sybase does not recommend installing and upgrading versions 15.5 to 15.7.1 SP100 using existing directories containing any of preupgrade versions 15.5 to 15.7.1 SP100. You must back up the files and directories of the preupgrade version of Replication Server before you install the new version of Replication Server in the same directories as the current version of Replication Server.

The procedures in *Upgrading Using Existing Directories* on page 83 and *Downgrading to an Earlier Version in Existing Directories* on page 85 only apply between Replication Server versions:

- 15.5 and 15.7
- 15.6 and 15.7
- 15.5 and 15.7.1
- 15.6 and15.7.1
- 15.7 and 15.7.1
- 15.5 and 15.7.1 SP100
- 15.6 and 15.7.1 SP100
- 15.7 and 15.7.1 SP100
- 15.7.1 and 15.7.1 SP100

Upgrading Using Existing Directories

Upgrade using the existing directories.

1. Shut down the preupgrade version of Replication Server.

   See *Start or Stop a Replication Server* on page 97.
2. Back up the entire set of files and directories under $SYBASE.

   Note: Keep this backup in case you need to downgrade from Replication Server version 15.7.1 SP100 to an earlier version.
3. Use the relevant version of the *Replication Server Installation Guide for UNIX* to install your new version of Replication Server into the existing version directories under $SYBASE.

> **Note:** Do not install the SAMPLE_RS sample Replication Server from the new version if you are using the SAMPLE_RS from the earlier version.

The installation process:

- Updates all subdirectories of `$SYBASE` that share directory names in both versions.
- Backs up existing `SYBASE.*` files, where `*` can be `.env`, `.sh`, or `.csh`, by adding a "save" suffix to the file. For example, `Sybase.env` is backed up as `Sybase.env.save`

4. Set the relevant environment variables in the `SYBASE.env` file in the Replication Server 15.7.1 SP100 installation directory.

   See *Replication Server Installation Guide > Postinstallation Tasks > Environment Variables on UNIX*.

5. You can use the **repserver** or **rs_init** utilities to upgrade Replication Server from versions 15.5 and later to version 15.7.1 SP100.

   However, to upgrade from versions 15.5, 15.6 and later to version 15.7, you can only use **rs_init**.

   For example, to upgrade the tokyo_rs Replication Server 15.6 ESD #1 using **repserver**, enter:

```
/sybase/rs1560esd1/REP-15_5/bin/repserver -upgr -Stokyo_rs
-C/sybase/rs1560esd1/REP-15_5/install/tokyo_rs.cfg
-E/sybase/rs1560esd1/REP-15_5/install/tokyo_rs.log
-I/sybase/rs1560esd1/interfaces
-A /sybase/rs1560esd1/REP-15_5/ASA11
```

   After the upgrade, you see a new entry in the configuration file. For example after upgrading tokyo_rs, you see:

```
erssd_release_dir=/sybase/rs1560esd1/REP-15_5/ASA11
erssd_start_cmd=/sybase/rs1560esd1/REP-15_5/ASA11/bin/dbspawn
-f -q /sybase/rs1560esd1/REP-15_5/ASA11/bin/dbsrv11
-ti 0 -x "tcpip(PORT=11753;DOBROAD=NO;BLISTENER=NO)"
-o /sybase/rs1560esd1/REP-15_5/errorlog/tokyo_rs_erssd.out
/sybase/rs1560esd1/REP-15_5/dbfile/tokyo_rs_erssd.db
RS_random=b6BAdgNj4GOgnJ47vYwkgcPoQflYvQOxidNOsgnrqCgIk6tmrvYrWmF
20hcV/SnPRH
```

> **Note:** You see the **RS_random** encryption attribute only if you upgrade to version 15.7.1 and later either with **repserver** or **rs_init**. As earlier versions of Replication Server cannot use the attribute, you can leave it in the configuration file or delete it if you downgrade from versions 15.7.1 and later.

6. Change the runserver file you copied from the backup to use the upgraded Replication Server executable, and then start Replication Server.

   See *Starting a Replication Server Using the Runserver File* in the *Replication Server Configuration Guide for UNIX*.

Downgrading to an Earlier Version in Existing Directories

You can revert to an earlier Replication Server version in the existing directories if you backed up your earlier version before upgrading to same directories.

1. Back up the entire set of files and directories under `$SYBASE`.

   These contain the current Replication Server version files.

2. Downgrade Replication Server using the procedure in *Downgrading Replication Server* on page 77.

3. Back up the Replication Server ERSSD or RSSD after completing the downgrade procedure in step 2.

4. To overwrite the directory structure under `$SYBASE`, restore the files and directories from the backup you made before you upgraded to the current Replication Server version.

5. Verify that the `.cfg`, `interfaces`, and `run_repservername` runserver files are configured to run the version of Replication Server you are downgrading to.

6. If there was replication activity on the upgraded version of Replication Server prior to the downgrade process, verify that you are using the downgraded version of RSSD files, otherwise restore the ERSSD or RSSD from the backup you have made in step 3.

   If you use an ERSSD, copy the `dbfile`, `translog`, `errorlog`, and `backup` directories to the corresponding locations. If you use SAMPLE_RS, copy all files and directories under `$SYBASE/REP-15_5/samp_repserver` to the corresponding locations.

7. Set the relevant environment variables in the `SYBASE.env` file.

   See *Environment Variables on UNIX* in the *Replication Server Installation Guide*.

8. Start the downgraded Replication Server version with the runserver file.

   See *Starting a Replication Server Using the Runserver File* in the *Replication Server Configuration Guide for UNIX*.

Password Encryption

Replication Server encrypts all passwords, and stores and transmits passwords in encrypted format.

Replication Server uses password encryption instead of clear text when storing all passwords for new Replication Server installations.

Set the site version to 1571 or later before you use **rs_init** to change existing encrypted passwords when you are upgrading Replication Server. To change the passwords if you do not set the site version to 1571 or later, you must use **rs_init** from the version of Replication Server that you are upgrading from. You should manually remove the **RS_random** attribute in the configuration file if you downgrade Replication Server.

To regenerate the password encrytion key, see *Replication Server Administration Guide Volume 1 > Manage Replication Server Security > Manage Replication Server System Security > Password Encryption*.

Enabling Password Encryption for a Replication Server

Enable password encryption for a Replication Server using **rs_init**.

rs_init creates a new configuration file that contains encrypted passwords for the Replication Server. **rs_init** encrypts the passwords in the rs_users and rs_maintusers system tables.

1. Log in as the "sybase" user and change to the Sybase installation directory.
2. Start **rs_init**. Enter:

   ```
   $SYBASE/$SYBASE_REP/install/rs_init
   ```

   You see the RS_INIT menu.
3. Select **Configure a Server Product**.
4. Select **Replication Server**.
5. Select **Enable Password Encryption for a Replication Server**.
6. Enter:
   - Name of the Replication Server for which you want to enable password encryption.
   - Login name for the Replication Server with "sa" user.
   - Password for the Replication Server "sa" user.
   - Full path for the Replication Server configuration file.
7. Press **Ctrl**+**A** to save your changes. **rs_init** displays:

   ```
   Execute the Replication Server tasks now?
   ```

8. Enter "y" to enable encryption.

    rs_init shuts down the Replication Server and encrypts the passwords in the Replication Server configuration file and in the rs_users and rs_maintusers system tables.

    ```
    Replication Server 'rs_name' can now be restarted.
    ```

9. Press Return. **rs_init** displays:
    ```
    Configuration tasks completed successfully.
    ```

10. Press Return. You see the Configure Replication System screen.

11. Press **Ctrl+C** and enter "y" to exit **rs_init**.

12. Restart the Replication Server and any connected Replication Agents.

13. Delete the backup of the Replication Server configuration file.

Note: The backup file contains unencrypted passwords, so you should remove that file to protect your replication system security. However, you may first want to make a hard copy or offline backup of the file and store it in a safe place.

The backup file has the same name as the Replication Server configuration file, but the extension is changed to a three-digit number such as "001." The file with the highest extension number is the most recent backup file.

Altering an Encrypted Password for a Replication Server

Change an encrypted password in a Replication Server configuration file using **rs_init**.

Do not use **rs_init** to change the passwords stored in the RSSD. To change the Replication Server password, use the **alter user** command. To change the maintenance user password, use **alter connection** or **alter route**.

1. Log in as the "sybase" user and change to the Sybase installation directory.

2. Start **rs_init**. Enter:
    ```
    $SYBASE/$SYBASE_REP/install/rs_init
    ```
    You see the RS_INIT menu.

3. Select **Configure a Server Product**.

4. Select **Replication Server**.

5. Select **Alter a Replication Server Configuration File Password**.

6. Enter:
    - Name of the Replication Server whose configuration file you want to update.
    - Login name for the Replication Server "sa" user.
    - Password for the Replication Server "sa" account.
    - Full path name for the Replication Server configuration file.
    - Parameter for the password you want to alter:

- • *RSSD_primary_pw_enc* – for the RSSD primary user.
  - • *RSSD_maint_pw_enc* – for the RSSD maintenance user.
  - • *ID_pw_enc* – for the ID Server user name.
- • New password you want to use.

7. Press **Ctrl**+**A** to save your changes. **rs_init** displays:
   ```
   Execute the Replication Server tasks now?
   ```

8. Enter "y".

   rs_init shuts down the Replication Server, then records the new password in the Replication Server configuration file. When that process is complete, **rs_init** shows:

   ```
   Configuration completed successfully.
   ```

   ```
   Press Return to continue.
   ```

9. Press **Return**. You see the Configure Replication System screen.

10. Press **Ctrl**+**C** and type "y" to exit **rs_init**.

11. Restart the Replication Server.

Password Encryption

Secure Sockets Layer

Secure Sockets Layer (SSL) is an industry standard for sending wire- or socket-level encrypted data over secure network connections.

See *Replication Server Administration Guide Volume 1 > Manage Replication Server Security* for detailed information on the SSL security option.

See also
* *Replication Server Security Information* on page 17

Setting Up SSL Services

Set up SSL services on Replication Server.

Prerequisites
Review the SSL Plus user documentation and documentation for any third-party SSL security software you are using.

Task

1. *Include Trusted CA certificates.*

   Modify the Open Client/Server™ trusted roots file to include trusted CA certificates.
2. *Obtain a Certificate for Each SSL-Enabled Replication Server*

   Obtain a certificate from a trusted certificate authority (CA) for each Replication Server accepting SSL connections.
3. *Create an Identity File*

   Create the identity file that contains the concatenation of a certificate and its corresponding encrypted private key, and that is understood by the SSL Plus software.
4. *Add an Encrypted Password to the Configuration File*

   Use rs_init to enable SSL on Replication Server and to add or change an encrypted SSL password.
5. *Create SSL Entries in the Directory Service*

   Create SSL entries in the Replication Server directory service—such as the interfaces file or an LDAP server, which defines the server address and port numbers, and determines security protocols that are enforced for client connections.
6. *Enable or Disable SSL on Replication Server*

You can enable SSL using Replication Server commands or rs_init.

7. *Restart Replication Server After SSL Changes*

Stop and start Replication Server for SSL configuration changes to take effect.

Include Trusted CA certificates.

Modify the Open Client/Server™ trusted roots file to include trusted CA certificates.

The list of known and trusted CAs is maintained in the trusted roots file: `$SYBASE/config/trusted.txt`

The system administrator adds and deletes CAs using a standard ASCII-text editor. The trusted roots file is similar in format to a certificate file.

By default, Replication Server recognizes these third-party CAs:

- Thawte
- Entrust
- Baltimore
- VeriSign
- RSA

Obtain a Certificate for Each SSL-Enabled Replication Server

Obtain a certificate from a trusted certificate authority (CA) for each Replication Server accepting SSL connections.

The system administrator installs server certificates and private keys for the Replication Server installation. The system administrator can get a server certificate using third-party tools provided with an existing public-key infrastructure (PKI) already in use in the customer environment. Each Replication Server acting as a server in an SSL-enabled connection must have a certificate issued by a trusted CA.

Most third-party PKI vendors have utilities to generate certificates and private keys.

If you request a certificate from a third-party vendor, and that certificate is in PKCS #12 format, use the **certpk12** utility to convert the certificate into a format understood by Replication Server.

Adaptive Server Enterprise provides the **certreq** and **certauth** utilities to test the certificate request tool and to verify that the authentication methods are working on your server. See *Adaptive Server Enterprise System Administration Guide > Security Administration*.

Creating an Identity File

Create the identity file that contains the concatenation of a certificate and its corresponding encrypted private key, and that is understood by the SSL Plus software.

Prerequisites
Verify that you have the system administrator role before performing this task.

Task

The name and default location of the identity file is the following, where *servername* is the name of the server as specified at start-up:

```
$SYBASE/$SYBASE_REP/certificates/servername.crt
```

1. To place the identity file in a different location, specify the alternate location in the **RS_ssl_identity** entry in the configuration file.
2. To make a successful connection, the common name in the certificate must match the Replication Server name in the directory service.

   Client certificates are not supported.

Adding an Encrypted Password to the Configuration File

Use **rs_init** to enable SSL on Replication Server and to add or change an encrypted SSL password.

The SSL password decodes the identity file private keys. This password is stored as the **RS_ssl_pw** entry in the Replication Server configuration file where it will not be sent over the network.

Note: You can enable and disable SSL on Replication Server using **configure replication server** and the **use_ssl** option.

Creating SSL Entries in the Directory Service

Create SSL entries in the Replication Server directory service—such as the `interfaces` file or an LDAP server, which defines the server address and port numbers, and determines security protocols that are enforced for client connections.

Replication Server implements the SSL protocol as a filter that is appended to master and query lines in the directory service.

1. Verify that all attempts to connect to a master or query entry in a directory service with an SSL filter supports the SSL protocol.

   For example, suppose a primary Replication Server (SYBSRV1) and a replicate Replication Server (SYBSRV2) use the Transmission Control Protocol (TCP) and the SSL protocol for communication.

   Entries in the `interfaces` file might look like this:

   ```
   SYBSRV1
   query tcp myserver sybasehost1 5001 ssl
   master tcp myserver sybasehost1 5001 ssl
   ```

   ```
   SYBSRV2
   query tcp myserver sybasehost2 4001 ssl
   master tcp myserver sybasehost2 4001 ssl
   ```

2. You can also configure Replication Server to accept SSL connections and, at the same time, have other connections that accept clear text or use security mechanisms such as DCE and Kerberos.

To support both SSL and other connection protocols, you must use multiple `interfaces` files.

For example, a typical `interfaces` file entry that supports the TCP and both SSL and clear text connections looks like this:

```
SYBSRV1
query tcp myserver sybasehost1 5001 ssl
master tcp myserver sybasehost1 5001 ssl
master tcp myserver sybasehost1 5001
```

3. The `interfaces` file master line entries allow SYBSRV1 to listen for both SSL and clear text connections. To make sure that SYBSRV1 sends queries to SYBSRV2 using SSL, there must be a single query entry in the `interfaces` file for SYBSRV1.

To allow SYBSRV1 to send queries to other servers using a different protocol, use a separate `interfaces` file.

Enabling or Disabling SSL on Replication Server

You can enable SSL using Replication Server commands or **rs_init**.

Note: Replication Server does not require the REP_SSL license, as SSL comes as a part of the basic REP_SERVER license.

---

1. Enable or disable SSL using **configure replication server** with the **use_ssl** option.

To use **configure replication server**, enter:

```
configure replication server
set use_ssl to 'on'
```

Set **use_ssl** to off to disable SSL. By default, SSL is not enabled on Replication Server. When **use_ssl** is off, Replication Server does not accept SSL connections.

use_ssl is a static option. You must restart Replication Server after you change its value.

2. To log in to Replication Server running on IBM AIX 64-bit platforms, edit the LIBPATH environment variable in SYBASE.csh or SYBASE.sh.

File	Execute
SYBASE.csh	`source SYBASE.csh` `setenv LIBPATH $SYBASE/$SYBASE_OCS/lib3p:$LIBPATH`
SYBASE.sh	`source SYBASE.sh` `export LIBPATH=$SYBASE/SYBASE_OCS/lib3p:$LIBPATH`

Restarting Replication Server After SSL Changes

Stop and start Replication Server for SSL configuration changes to take effect.

See also

*   *Start or Stop a Replication Server* on page 97

Secure Sockets Layer

Start or Stop a Replication Server

You can start a Replication Server using the `runserver` file created by **rs_init** and connect using **isql**, then executing **shutdown** to stop.

Start-up Sequence for Servers

In a production system, you must start Adaptive Servers and Replication Servers, whenever your machine reboots. See your system administrator to do this.

Start servers in this order:

1. If the Replication Server uses RSSD in Adaptive Server Enterprise, start the Adaptive Server first.
2. Replication Servers.

Runserver File

The runserver file is an executable script that contains the complete command line required to start a Replication Server. When you install a new Replication Server to your replication system, **rs_init** creates a runserver file in the Sybase installation directory.

The runserver file name is constructed based on the name of the server. For example, for a Replication Server named ROME_RS, the runserver file is named `RUN_ROME_RS`.

Starting a Replication Server Using the Runserver File

Use the runserver file created by **rs_init** to start a Replication Server.

1. Log in to your system as the "sybase" user. Replication Server must execute as "sybase" so that it has the permissions required to access its configuration file, log file, and disk partitions.
2. Set the $SYBASE_REP and $SYBASE_OCS environment variables to the path of the Sybase installation directory, as described in the *Replication Server Installation Guide*.
3. From the shell prompt, execute the runserver file as a background process.

   For example, enter this command to start the TOKYO_RS Replication Server:

   ```
   $SYBASE/$SYBASE_REP/RUN_TOKYO_RS &
   ```

Diagnostic messages appear as the servers start. The same messages are also recorded in the servers' log files.

Starting Replication Server with Sufficient Memory

Configure Replication Server or the operating system to provide sufficient memory when Replication Server starts up.

With sufficient memory, Replication Server can avoid failure when there is heavy memory consumption by Replication Server caches and buffers.

1. Modify the **ulimit** option settings in the `runserver` file to:

```
ulimit -d 'unlimited'
• ulimit -c 'unlimited'
• ulimit -m 'unlimited
```

2. (Optional) Modify the operating system memory limits.

   For example, on IBM AIX, change the default settings in the `/etc/security/limits` file:
   - fsize – 1
   - core – 2097151
   - cpu – 1
   - data – 1
   - rss – 1
   - stack – 65536
   - nofiles – 2000

   See your operating system documentation for similar implementations

Stopping a Replication Server Using isql

Stop a Replication Server using the **isql** utility.

When you shut down a Replication Server, it refuses additional connections, terminates threads, and exits.

1. Use **isql** to log in to the Replication Server as the System Administrator:

```
isql -Usa -Psa_password -Sservername
```

2. Enter:

```
shutdown
go
```

Replication Monitoring Services

The Replication Manager (RM) plug-in for Sybase Central can manage small and simple replication environments with fewer than ten servers. However, to manage large and complex replication environments, the RM can use a management server layer called Replication Monitoring Services (RMS).

Replication Monitoring Services (RMS) monitors the servers and components in a replication environment, provides the ability to control the flow of data in the replication environment, and sets the configuration parameters.

RMS functionality is available through the RM or a command line API. RM provides commands to create, modify, or delete replication objects; RMS provides an API to monitor and troubleshoot the replication environment.

RMS is applicable only for three-tier management support. In three-tier management support, the RM connects to the servers in the environment through the RMS. RMS provides the monitoring capabilities for the replication environment.

RMS monitors the status of the servers and other components in the replication environment, and the RM provides the client interface that displays the information provided by the RMS.

Using RMS, you can monitor these servers in your replication domain:

- Adaptive Server Enterprise
- DirectConnect™

  Note: DirectConnect for Oracle is a component of ECDA for Oracle. With Replication Server Options 15.5, ExpressConnect for Oracle replaced ECDA for Oracle.

- Mirror Replication Agent
- Open Server™
- Other RMS Servers
- Replication Agent
- Replication Server
- Sybase IQ – you must use the RMS API for Sybase IQ. See *Replication Server Reference Manual > Replication Monitoring Services API*.

You can download the RMS installer from the Sybase Web site under the Replication Server product.

The RMS component is automatically installed in a full or typical installation.

Starting RMS

Start Replication Monitoring Services.

1. Navigate to the RMS installation directory, $SYBASE.
2. Set the environment variables by sourcing SYBASE.csh
3. Change to the $SYBASE/SCC-3_2/bin directory.
4. Execute **scc.sh**

Stopping RMS

You can use commands to stop RMS.

1. Navigate to the Replication Server installation directory, $SYBASE.
2. Set the environment variables by sourcing SYBASE.csh
3. Change to the $SYBASE/SCC-3_2/bin directory.
4. Execute:

```
scc.sh -stop
```

Configuring RMS for Three-Tier Management Support

Configure RMS to support a three-tier environment.

1. Navigate to the $SYBASE directory and source SYBASE.csh
2. Add an entry in the $SYBASE/interfaces file for each server you want to monitor
   with RMS. (Optional) To use RMS with Replication Manager, add an entry for RMS.
   • Server Name – **RMServices**
   • Port – 32008
   • Hostname – host name of the machine on which RMS will run
3. To start RMS, go to the $SYBASE/SCC-3_2/bin directory and run scc.sh
4. Add an entry for RMS in the interfaces file of the host where Replication Server is
   running:
   • Server Name – **RMServices**
   • Port – 32008
   • Hostname – host name of the machine on which RMS will run

5. To configure the RMS server, start an **isql** session by entering:

```
isql -Usa -Psa_pwd -SRMServices
```

6. To change the RMS password, enter:

```
configure password=newPassword
go
```

7. You can now start using Replication Manager or the RMS API commands to monitor a replication environment. For example:

```
add replication server SAMPLE_RS set username to sa
```

See *Replication Server Reference Manual > Replication Monitoring Services API* for API definitions.

Connecting to RMS

Connect to RMS using the RM plug-in once it has been started. You can also use the RM plug-in to add servers that need to be monitored by the RMS.

1. Click the **Connect** icon from the toolbar to open the Connect to a Replication Domain window.
2. Select **RMS Server**.
3. Enter the user name and password needed to connect to RMS.
4. Select **RMS** from the list of servers in the drop-down list, or click the **Options** button to provide the connection information for the RMS.
5. Enter a server name, host, and port number.
6. Click **OK**. The RMS server is added to your object tree.

See *Replication Server Reference Manual > Replication Monitoring Services API*.

Install and Implement Non-ASE Support Features

There are several requirements for the installation and implementation of non-ASE support features.

Replication Server support for replication into a non-ASE replicate database requires:

- Replication Server system tables—rs_lastcommit and rs_info— created in the non-ASE database, and Replication Server support functions such as **rs_get_lastcommit** and **rs_get_charset**.
- Function strings that:
  1. Replace certain language command and stored procedure invocations from the replicate Replication Server, then,
  2. Produce operations in the replicate database that emulate the behavior of the commands and stored procedures that would exist in a Sybase Adaptive Server Enterprise replicate database.
- Heterogeneous datatype support for non-ASE datatypes that do not map directly to Replication Server datatypes (or Sybase ASE datatypes), which must be translated to the appropriate datatypes for replication into the replicate database.

Components for Non-ASE Support Installed with Replication Server

The components included in the Replication Server default installation are connection profiles that contain function-string classes and error classes for each actively supported database, datatypes class, Replication Server native datatype literal descriptor row in the RSSD rs_datatype table, and Replication Server datatype translation engine.

Replication Server includes connection profiles that you use to connect to non- ASE data servers. Connection profiles contain or install these components:

- Datatype definitions – assign a unique datatype ID to each datatype, and describe the attributes of each non-ASE datatype, such as delimiters, length, scale, precision, minimum and maximum values, how it is represented using native Replication Server datatypes. Replication Server uses datatype definitions to:
  - Identify datatypes
  - Convert datatypes into native datatypes for internal handling (if necessary)
  - Translate datatypes
- Replication Server support objects in the replicate database – Replication Server relies on a replicate database to maintain some of the operational data that Replication Server needs to replicate to that database.

Non-ASE support accommodates the Replication Server replicate database requirements by creating two tables in the replicate database:

- `rs_info` – stores the character set and sort order used by the database. The Replication Server maintenance user ID must have select permission on this table.
- `rs_lastcommit` – is used by the replicate Replication Server to record information about the most recent replicated operations committed in the replicate database. Because this table is both read and updated by the replicate Replication Server, the Replication Server maintenance user ID must have both insert and update permission on this table.

- Replication Server function strings – allow users to customize the database operations sent by the replicate Replication Server to the replicate database. Non-ASE support uses function strings to replace invocations of stored procedures and certain database commands that the replicate Replication Server sends to the replicate database.

  The function strings required to support connections to non-ASE data servers are installed in the RSSD of the replicate Replication Server.

  For functions-string classes, see *Replication Server Administration Guide Volume 2 > Customizing Database Operations* and for datatype classes, see *Replication Server Reference Manual > Topics*.

- Class-level translations for non-ASE datatypes – default datatype-to-datatype mappings, provided by connection profiles.

  Connection profiles provide a function-string class for each supported database server. The class-level translations are specific to each function-string class that supports a non-Sybase replicate database.

Setting Up and Configuring Non-ASE Support

Set up and configure non-ASE support in Replication Server.

1. Create a replicate database connection – use the **using profile** clause with the **create connection** command to create a connection to a non-ASE replicate database. See *Replication Server Reference Manual > Replication Server Commands > **create connection using profile***.
2. Create replicate definitions – see *Replication Server Administration Guide Volume 1 > Manage Replicated Tables > Create Replication Definitions*.
3. Mark tables for replication – see *Replication Server Administration Guide Volume 1 > Manage Replicated Tables > Mark Tables for Replication*.
4. Create subscriptions – see *Replication Server Administration Guide Volume 1 > Manage Subscriptions*.

Once you complete all these steps, you can begin replicating transactions to a non-ASE replicate database. See the *Replication Server Heterogeneous Replication Guide* for information specific to supported non-ASE databases.

Sample Replication System

The sample replication system replicates the changes made to two tables in the `pubs2` database from one Adaptive Server to another Adaptive Server.

You can set up this example as an introduction to using Replication Server or to test your replication system installation.

Note: The application architecture for this example is similar to the decision-support model described in the *Replication Server Design Guide*.

The goal of this replication system is to replicate all changes made to the `authors` and `titles` tables in the `pubs2` database in one Adaptive Server to the same tables in the `pubs2` database in another Adaptive Server.

The replication system has two Adaptive Servers:

- A primary Adaptive Server, named TOKYO_DS, holds the RSSD for the primary Replication Server and the primary `pubs2` database.
- The replicate Adaptive Server, named SYDNEY_DS, holds the RSSD for the replicate Replication Server and the replicate `pubs2` database.

The replication system has two Replication Servers:

- A primary Replication Server named TOKYO_RS manages the primary `pubs2` database.
- A replicate Replication Server named SYDNEY_RS manages the replicate `pubs2` database.

The replication system has two RepAgents, which are the Replication Agents for Adaptive Server:

- A RepAgent for the primary RSSD runs in the ASE TOKYO_RSSD and transfers the RSSD log to the primary Replication Server.
- A RepAgent for the primary `pubs2` database runs in the ASE TOKYO_PUBS2 and transfers the `pubs2` log to the primary Replication Server.

Setting Up the Sample Replication System

Use the Adaptive Servers or Replication Servers that you have already installed in setting up the sample replication system. Substitute the names of your servers throughout the instructions.

Prerequisites

Choose the machines where each server will execute and locate disk space for Adaptive Server database devices and Replication Server partitions. See the *Replication Server Installation Guide* for disk space and system requirements.

Task

1. Install the primary Adaptive Server, TOKYO_DS.

2. Install the replicate Adaptive Server, SYDNEY_DS.

3. Install the `pubs2` database in the primary Adaptive Server:

   ```
   isql -Usa -Psa_passwd -STOKYO_DS < \
   $SYBASE/$SYBASE_ASE/scripts/installpubs2
   ```

4. Install the `pubs2` database in the replicate Adaptive Server:

   ```
   isql -Usa -Psa_passwd -SSYDNEY_DS < \
   $SYBASE/$SYBASE_ASE/scripts/installpubs2
   ```

5. Complete the Replication Server Installation worksheet for the primary Replication Server (TOKYO_RS).

   • Create the RSSD for TOKYO_RS in the TOKYO_DS Adaptive Server.
   • The RSSD requires a RepAgent.

6. Complete the Replication Server Installation worksheet for the replicate Replication Server (SYDNEY_RS).

   • Create the RSSD for SYDNEY_RS in the SYDNEY_DS Adaptive Server.
   • The RSSD does not require a RepAgent.

7. Complete the Database Setup worksheet for the primary `pubs2` database.

   • The database is in Adaptive Server TOKYO_DS.
   • The database is managed by the TOKYO_RS Replication Server.
   • The database requires a RepAgent.

8. Complete the Database Setup worksheet for the replicate `pubs2` database.

   • The database is in Adaptive Server SYDNEY_DS.
   • The database is managed by the SYDNEY_RS Replication Server.
   • The database does not require a RepAgent.

9. Log in to the machine where the primary Replication Server will execute. Run **rs_init** and install Replication Server TOKYO_RS using the worksheet you completed in step 5.

10. Log in to the machine where you installed the primary Replication Server. Run **rs_init** and add the primary pubs2 database to the replication system using the worksheet you completed in step 7.

11. Log in to the machine where the replicate Replication Server will execute. Run **rs_init** and install Replication Server SYDNEY_RS using the worksheet you completed in step 6.

12. Log in to the machine where you installed the replicate Replication Server. Run **rs_init** and add the replicate pubs2 database to the replication system using the worksheet you completed in step 8.

See also
*   *Replication Server Installation Worksheet Sample* on page 10
*   *Database Setup Worksheet Sample* on page 26

Establishing the Sample Replication

Create a route from the primary Replication Server to the replicate Replication Server to establish a sample replication.

1. Log in to the primary Replication Server as "sa":
```
isql -Usa -STOKYO_RS
```

2. Enter the **create route** command:
```
create route to SYDNEY_RS
   set username SYDNEY_RS_rsi
   set password SYDNEY_RS_rsi_ps
go
```

Note: Find the user name and password for the **create route** command in the "Remote site connections" section of the Replication Server Installation Worksheet for the replicate Replication Server.

3. Create a login account in the primary Adaptive Server for the user who will create the replication definitions and subscriptions. In this example, the login name is "repsys." Add the user to the pubs2 databases and grant the user select permissions for the tables to replicate.
```
isql -Usa -STOKYO_DS
      sp_addlogin repsys, repsys_ps
      go

      use pubs2
      go

      sp_adduser repsys
      go
```

```
        grant select on authors to repsys
        grant select on titles to repsys
        go
        quit
```

4. Add the "repsys" user to the two Replication Servers and grant create object permission to the user. The "repsys" user must have the same login name and password in the two Replication Servers and the primary Adaptive Server:

```
isql -Usa -STOKYO_RS
        create user repsys
        set password repsys_ps
        go

        grant create object to repsys
        go
        quit

isql -Usa -SSYDNEY_RS
        create user repsys
        set password repsys_ps
        go

        grant create object to repsys
        go
        quit
```

5. Create replication definitions for the authors and titles tables:

```
isql -Urepsys -Prepsys_ps -STOKYO_RS
        create replication definition authors
        with primary at TOKYO_DS.pubs2
        with all tables named 'authors'
          (
            au_id varchar(11),
            au_lname varchar(40),
            au_fname varchar(20),
            phone char(12),
            address varchar(40),
            city varchar(20),
            state char(2),
            country varchar(12),
            postalcode char(10)
          )
        primary key (au_id)
        searchable columns (state, postalcode)
        go
        create replication definition titles
        with primary at TOKYO_DS.pubs2
        with all tables named 'titles'
        (
            title_id varchar(6),
            title varchar(80),
            type char(12),
            pub_id char(4),
            price money,
            advance money,
```

```
       total_sales int,
       notes varchar(200),
       pubdate datetime,
       contract bit
)
primary key (title_id)
searchable columns (type, pub_id)
go
```

6. Set replication to on for the authors and titles tables in the primary pubs2 database:

```
isql -Usa -STOKYO_DS
     use pubs2
     go

     sp_setreptable authors, true
     go

     sp_setreptable titles, true
     go
```

7. In the replicate pubs2 database, grant permissions on the authors and titles tables to the maintenance user:

```
isql -Usa -SSYDNEY_DS
     use pubs2
     go

     grant select, insert, delete, update
     on authors to pubs2_maint
     grant select, insert, delete, update
     on titles to pubs2_maint
     go
```

Note: You can find the maintenance user in the "Database information" section of the Database setup worksheet you completed for the replicate pubs2 database.

8. In the replicate Replication Server, create subscriptions for the authors and titles tables:

```
isql -Urepsys -Prepsys_ps -SSYDNEY_RS
     create subscription authors_sub
     for authors
     with replicate at SYDNEY_DS.pubs2
     without materialization
     go

     create subscription titles_sub
     for titles
     with replicate at SYDNEY_DS.pubs2
     without materialization
     go
```

> **Note:** Since there is already data at the replicate database, this example uses the **create subscription** with the **without materialization** option. For other methods, see *Replication Server Reference Manual > Replication Server Commands.*

Troubleshooting and Testing the Replication System

Troubleshoot and test replication for the `authors` and `titles` tables once it has been established.

If you created a subscription with materialization, log in to the replicate `pubs2` database and select the data from the `titles` and `authors` tables:

- If the rows exist in the replicate tables – it indicates that the subscriptions were created and materialized successfully.
- If the rows do not appear in the replicate tables – execute **check subscription** in each of the Replication Servers:

```
check subscription authors_sub
for authors
with replicate at SYDNEY_DS.pubs2
```

  check subscription reports the status of the subscription. If the status is not "valid" in both the primary and replicate Replication Servers, then either the subscription has not finished materializing or an error has occurred.

The `authors` and `titles` tables are very small. If there are no configuration problems, they should not take more than a few seconds to materialize. If you do not see the results quickly, you can assume that an error has occurred.

If you created a subscription without materialization or the subscription has completed materializing, log in to the primary `pubs2` database and execute some **insert**, **update**, and **delete** SQL commands against the `authors` and `titles` tables. Then log in to the replicate `pubs2` database and verify that your modifications appear in the replicate tables. If the rows do not appear in the replicate tables:

- Execute **check subscription** in each of the Replication Servers.
- Check the error logs for the primary and replicate Replication Servers for error messages. The most common problems are:
  - Failure to log in to the primary Adaptive Server. The user who creates the subscription in the replicate Replication Server must have the same login name and password in the primary Adaptive Server and the primary Replication Server.
  - Missing permissions in the primary database. The user who creates the subscription must be a user in the primary database and must have select permission in the primary table.
  - Missing permissions in the replicate database. The maintenance user must have select, insert, update, and delete permissions on the tables in the replicate database.

- A Replication Server or Adaptive Server has stopped running. Try logging in to each of the servers. Restart any servers that are not running.

Index

Index