SYBASE®

Installation Guide

Contents

vi

About This Book

This guide, *Installation Guide Sybase Adaptive Server Enterprise on Windows NT* provides:

- An overview of the Sybase® Adaptive Server™ Enterprise installation infrastructure

- Instructions for installing and upgrading Adaptive Server, and installing Backup Server™, Monitor Server, XP Server™, jConnect™ for JDBC™, Java utilities, and client products, including the Adaptive Server plug-in for Sybase Central™

- Instructions for installing optional Adaptive Server functionality, such as auditing, and sample databases, and localization information

Audience

This guide is written for Sybase System Administrators and other qualified personnel who are familiar with their system's environment, networks, disk resources, and media devices.

How to use this book

This book contains the following chapters:

- Chapter 1, "Overview" provides product definitions and an overview of the installation process and the Adaptive Server installation infrastructure.

- Chapter 2, "Installation Requirements" provides descriptions and system-related information such as required RAM and disk space for all Adaptive Server server and client products for this release.

- Chapter 3, "Sybase Software Asset Management (SySAM)" provides installation instructions for Sybase Software Manager (SySAM).

- Chapter 4, "Installing Sybase Servers" describes pre-installation procedures, and how to install server components.

- Chapter 5, "Post-Installation Tasks" describes how to set environment variables, install sample databases, and initialize features.

- Chapter 6, "Installing Sybase PC-Client Products" describes how to install Adaptive Server client products like the Adaptive Server plug-in to Sybase Central.

- Chapter 7, "Starting and Stopping Servers" describes the different methods of starting and stopping Adaptive Server, Backup Server, and other servers.

- Chapter 8, "Upgrading Sybase Servers" describes how to upgrade an existing Adaptive Server to the current version.

- Chapter 9, "Removing Sybase Servers" describes how to uninstall Sybase servers.

- Chapter 10, "Troubleshooting" provides installation error messages and possible solutions to installation problems.

- Appendix A, "Alternative Installation Methods" describes how to install Sybase servers in noninteractive mode.

Related documents   The following documents comprise the Sybase Adaptive Server Enterprise documentation:

- The release bulletin for your platform – contains last-minute information that was too late to be included in the books.

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

- The *Installation Guide* for your platform (this book)– describes installation, upgrade, and configuration procedures for all Adaptive Server and related Sybase products.

- *Configuring Adaptive Server Enterprise* for your platform – provides instructions for performing specific configuration tasks for Adaptive Server.

- *What's New in Adaptive Server Enterprise?* – describes the new features in Adaptive Server version 12.5, the system changes added to support those features, and the changes that may affect your existing applications.

- *Transact-SQL User's Guide* – documents Transact-SQL, Sybase's enhanced version of the relational database language. This manual serves as a textbook for beginning users of the database management system. This manual also contains descriptions of the pubs2 and pubs3 sample databases.

- *System Administration Guide* – provides in-depth information about administering servers and databases. This manual includes instructions and guidelines for managing physical resources, security, user and system databases, and specifying character conversion, international language, and sort order settings.

- *Reference Manual* – contains detailed information about all Transact-SQL commands, functions, procedures, and datatypes. This manual also contains a list of the Transact-SQL reserved words and definitions of system tables.

- *Performance and Tuning Guide* – explains how to tune Adaptive Server for maximum performance. This manual includes information about database design issues that affect performance, query optimization, how to tune Adaptive Server for very large databases, disk and cache issues, and the effects of locking and cursors on performance.

- The *Utility Guide* – documents the Adaptive Server utility programs, such as isql and bcp, which are executed at the operating-system level.

- The *Quick Reference Guide* – provides a comprehensive listing of the names and syntax for commands, functions, system procedures, extended system procedures, datatypes, and utilities in a pocket-sized book. Available only in print.

- The *System Tables Diagram* – illustrates system tables and their entity relationships in a poster format. Available only in print.

- *Error Messages and Troubleshooting Guide* – explains how to resolve frequently occurring error messages and describes solutions to system problems frequently encountered by users.

- *Component Integration Services User's Guide* – explains how to use the Adaptive Server Component Integration Services feature to connect remote Sybase and non-Sybase databases.

- *Java in Adaptive Server Enterprise* – describes how to install and use Java classes as datatypes, functions, and stored procedures in the Adaptive Server database.

- *Using Sybase Failover in a High Availability System* – provides instructions for using Sybase's Failover to configure an Adaptive Server as a companion server in a high availability system.

- *Using Adaptive Server Distributed Transaction Management Features* – explains how to configure, use, and troubleshoot Adaptive Server DTM features in distributed transaction processing environments.

- *EJB Server User's Guide* – explains how to use EJB Server to deploy and execute Enterprise JavaBeans in Adaptive Server.

- *XA Interface Integration Guide for CICS, Encina, and TUXEDO* – provides instructions for using Sybase's DTM XA interface with X/Open XA transaction managers.

- *Glossary* – defines technical terms used in the Adaptive Server documentation.

- *Sybase jConnect for JDBC Programmer's Reference* – describes jConnect for JDBC and explains how to use it to access data stored in relational database management systems.

- *Full-Text Search Specialty Data Store User's Guide* – describes how to use the Full-Text Search feature with Verity to search Adaptive Server Enterprise data.

- *Historical Server User's Guide* – describes how to use Historical Server to obtain performance information Adaptive Server.

- *Monitor Server User's Guide* – describes how to use Monitor Server to obtain performance statistics from Adaptive Server.

- *Monitor Client Library Programmer's Guide* – describes how to write Monitor Client Library applications that access Adaptive Server performance data.

Other sources of information

Use the Sybase Technical Library CD and the Technical Library Product Manuals Web site to learn more about your product:

- Technical Library CD contains product manuals and is included with your software. The DynaText browser (downloadable from Product Manuals at http://www.sybase.com/detail/1,6904,1010663,00.html) allows you to access technical information about your product in an easy-to-use format.

  Refer to the *Technical Library Installation Guide* in your documentation package for instructions on installing and starting the Technical Library.

- The Technical Library Product Manuals Web site is an HTML version of the Technical Library CD that you can access using a standard Web browser. In addition to product manuals, you will find links to the Technical Documents Web site (formerly known as Tech Info Library), the Solved Cases page, and Sybase/Powersoft newsgroups.

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

Sybase certifications on the Web   Technical documentation at the Sybase Web site is updated frequently.

❖ **For the latest information on product certifications**

1   Point your Web browser to Technical Documents at
    http://www.sybase.com/support/techdocs/.

2   Select Products from the navigation bar on the left.

3   Select a product name from the product list.

4   Select the Certification Report filter, specify a time frame, and click Go.

5   Click a Certification Report title to display the report.

❖ **For the latest information on EBFs and Updates**

1   Point your Web browser to Technical Documents at
    http://www.sybase.com/support/techdocs/.

2   Select EBFs/Updates. Enter user name and password information, if
    prompted (for existing Web accounts) or create a new account (a free
    service).

3   Specify a time frame and click Go.

4   Select a product.

5   Click an EBF/Update title to display the report.

❖ **To create a personalized view of the Sybase Web site (including support pages)**

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

1   Point your Web browser to Technical Documents at
    http://www.sybase.com/support/techdocs/

2   Click MySybase and create a MySybase profile.

Conventions   The following style conventions are used in this manual:

•   In a sample screen display, commands you should enter exactly as shown
    are given in:

    ```
    this font
    ```

•   In a sample screen display, words that you should replace with the
    appropriate value for your installation are shown in:

    ```
    this font
    ```

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

  /usr/u/sybase

- The names of programs, utilities, procedures, and commands appear in this font:

  sqlupgrade

- Commands for both the C shell and the Bourne shell are provided in this document, when they differ. The initialization file for the C shell is called *.cshrc*. The initialization file for the Bourne shell is called *.profile*. If you are using a different shell, such as the Korn shell, refer to your shell-specific documentation for the correct command syntax.

Table 1 shows the conventions for syntax statements in this manual.

Table 1: SQL syntax conventions

Key	Definition
command	Command names, command option names, utility names, utility flags, and other keywords are in bold.
variable	Variables, or words that stand for values that you fill in, are in *italic*.
{ }	Curly braces indicate that you choose at least one of the enclosed options. Do not include braces in your option.
[ ]	Brackets mean choosing one or more of the enclosed options is optional. Do not include brackets in your option.
( )	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 to be typed as part of the command.

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

Overview

This chapter introduces important concepts about the installation of Adaptive Server and Adaptive Server products.

Topic	Page
User roles	1
Product descriptions	2
Installation and setup utilities	4
PC-client product descriptions	5
Sybase installation directory	8

User roles

The Adaptive Server installation and setup process defines various user roles. Different user roles have different responsibilities and privileges. These user roles clarify the way Adaptive Server is to be integrated into your system:

- *Operating System Administrator* – the individual who maintains the operating system. This individual has superuser or "root" privileges.

- *Sybase System Administrator* – the individual in charge of Adaptive Server system administration, creating user accounts, assigning permissions on databases, and creating new databases. At installation time, the System Administrator's login name is "sa". The "sa" login is not a UNIX login. The "sa" login is specific to Adaptive Server and is used to log in to Adaptive Server using the isql command.

- *Sybase Login* – the "sybase" login is a UNIX login that owns all the Sybase installation directories and files, sets permissions on those directories and files, and performs the installation and upgrading of Adaptive Server.

Product descriptions

Table 1-1 provides a description of the server products in your standard Adaptive Server package.

Table 1-1: Product descriptions

Product	Description
Adaptive Server	The relational database server. The default unloading process includes: • Adaptive Server • Backup Server • Monitor Server • XP Server • Adaptive Server utilities • Scripts and configuration files
Adaptive Server Plug-In to Sybase Central Java Edition	Sybase Central is a common framework for managing servers. It allows you to manage Adaptive Server installations using the Sybase Central™ graphical management tool. For information on how to use Sybase Central to manage Adaptive Server, see the online help.
Sybase Monitor Server for Adaptive Server	An Open Server™ application that obtains performance statistics on Adaptive Server and makes those statistics available to Monitor Server client applications. Sybase Monitor Server includes: • Monitor Server for Adaptive Server Enterprise 12.5 – an Open Server application that obtains performance statistics on Adaptive Server and makes those statistics available to monitors Monitor Historical Servers and applications built with Monitor Client-Library. • Monitor Client-Library – a programming interface that provides access to Adaptive Server performance data. • Monitor Historical Server – an Open Server application that obtains performance statistics for many Adaptive Servers via Monitor Servers and records the data to specified file locations.

Product	Description
Backup Server	Backup Server is an Open Server-based application that manages all database backup (dump) and restore (load) operations for Adaptive Server. Backup Server: • Allows you to use virtually unlimited dump devices (this is called *dump striping*) in parallel to dump or load a single database or transaction log. • Allows compressed dumps and loads to the local disk. • Allows one dump to span multiple tapes or allows multiple dumps to be made to a single tape. • Allows dumping and loading over the network to or from a device on another machine. • Provides automated determination of tape device characteristics, by using operating system commands, for a dump operation. • Supports dump and load command syntax specification for volume naming, dismount and load control, tape density, block size, tape capacity, days to retain, initialization, file naming for multi-dump volumes, and listing header or file information. Install Backup Server if you plan to back up and restore databases in Adaptive Server. Backup Server is installed, by default, when you install Adaptive Server software.
Language Modules (Server)	Provides system messages and date/time formats to help you localize your applications. Default installation includes the us_english language module and the following character sets: • cp437 – IBM CP437, U.S. code set • cp850 – IBM CP850, European code set • iso_1 – ISO 8859-1, Latin-1 • mac – Standard Macintosh coding • roman8 – HP Roman-8 Other languages for Adaptive Server and Sybase client products are French, German, and Japanese. Language modules for Spanish, Korean, Brazilian Portuguese, and Simplified Chinese are also available for Adaptive Server only. For more information on server language modules and character sets, see *Configuring Adaptive Server Enterprise* for your platform.
Language Modules (Connectivity)	Provides messages and support files for running Open Client™ applications in various languages.
jConnect 4.5 and *jConnect 5.5*	Provides a Java database connectivity (JDBC) driver that works with both Sun and Microsoft virtual machines (VMs). Provides support for Adaptive Server 12.5 extended limits by requesting support for wide tables. This request is ignored by servers other than Adaptive Server 12.5 and higher. For more information on jConnect for JDBC, see the jConnect product page at http://www.sybase.com/support/manuals/.

Product	Description
jConnect Documentation	Contains the *Sybase jConnect for JDBC Programmer's Reference*.
Java utilities	Includes: • The Cascade Gateway – a gateway that acts as a proxy to provide a path to the database server if it is running on a different host from the Web server. • jisql – a graphical Transact-SQL® editor written in Java that replaces SQL Advantage®. • Ribo – a utility that captures, translates, and displays the Tabular Data Stream (TDS) protocol flowing between a TDS client and a TDS server.
ODBC Driver	Allows Windows NT client applications to access Adaptive Server data.
Open Client	Provides libraries and utilities for developing any Open Client application.
XP Server	An Open Server application that manages and executes extended stored procedures (ESPs) from within Adaptive Server. ESPs provide a method for calling procedural language functions from within Adaptive Server. XP Server is unloaded, by default, when you unload Adaptive Server software from the distribution media. Use the srvbuild utility to set up XP Server and to connect XP Server and Adaptive Server through the *interfaces* file.

Installation and setup utilities

Table 1-2 lists the Adaptive Server installation and setup utilities you can use to unload, install, upgrade, or customize Adaptive Server and its product.

Table 1-2: Installation and setup utilities

Utility	Use to	Environment
asecfg	Utility to access dsedit, svrbuild, sqlupgrade, sqlloc.	X-Windows/Motif GUI
dscp	Modify the *interfaces* file.	Interactive, text-based interface
dsedit	Modify the *interfaces* file.	X-Windows/Motif GUI
sqlloc	Modify localization settings, such as language and character set. For information on using sqlloc, see *Configuring Adaptive Server Enterprise*.	X-Windows/Motif GUI
sqllocres	Modify localization settings, such as language and character set, using a resource file. Use of this utility is not described in this installation guide. For information on using sqllocres, see *Configuring Adaptive Server Enterprise*.	Noninteractive, file-based interface
sqlupgrade	Upgrade Adaptive Server.	X-Windows/Motif GUI

Utility	Use to	Environment
sqlupgraderes	Upgrade Adaptive Server using a resource file.	Noninteractive, file-based interface
srvbuild	Build a functional server.	X-Windows/Motif GUI
srvbuildres	Build a functional server using a resource file.	Noninteractive, file-based interface
Studio Installer	• Provide an interface from which to select the installation type and setup task you want to perform. • Installs Adaptive Server products on to the hard drive.	X-Windows/Motif GUI

PC-client product descriptions

Table 1-3 describes Adaptive Server client products, which can be installed on Windows 98 and Windows NT client computers.

Table 1-3: Products and platforms

Product and platforms	Description
Open Client version 12.5	Used to develop and deploy C language-based applications that access Adaptive Server data. Both 32-bit and 64-bit libraries are shipped for 64-bit platforms. Default installation includes: • Client-Library™ • DB-Library • CS-Library • Bulk Library • bcp utility used to transfer data between files and Adaptive Server databases • isql utility used to connect to Adaptive Server for SQL queries. • defncopy utility used to copy stored procedure definitions from databases to files and back • Online help files for Open Client routines • Net-Library drivers for connecting 32-bit and 64-bit client applications to a server through most available network protocols. Default installation includes: 　• dsedit and dscp for editing the *sql.ini* and *interfaces* files and testing client-server connections 　• Drivers for Named Pipes and Windows Sockets 　• Drivers for Named Pipes, Windows Sockets, and Microsoft TCP/IP Additional options include: • Programmer support (include files and libraries) for compiling and linking Client-Library and DB-Library applications • Sample programs for Client-Library and DB-Library • Net-Library driver protocols: 　TCP/IP, SPX/IPX, Named Pipes, DECNet. 　TCP/IP and SPX/IPX require the WinSock API. DECNet requires DEC PathWorks. • Monitor Client Library – an application programming interface that provides access to Adaptive Server performance data
Language modules version 12.5	Display system messages and datetime formats to help you localize your 32-bit or 64-bit applications. The Standard Installation option installs the character sets.

Product and platforms	Description
Embedded SQL/C version 12.5	A superset of Transact-SQL that lets you place Transact-SQL statements in application programs written in languages such as C.
Embedded SQL/COBOL version 12.5	A superset of Transact-SQL that lets you place Transact-SQL statements in application programs written in languages such as COBOL.
jConnect 4.5 and *jConnect 5.5*	Provides a Java database connectivity (JDBC) driver that works with both Sun and Microsoft virtual machines (VMs).
	Provides support for Adaptive Server 12.5 extended limits by requesting support for wide tables. This request is ignored by servers other than Adaptive Server 12.5 and higher.
	For more information on jConnect for JDBC, see the jConnect product page at http://www.sybase.com/support/manuals/.
jutils (Cascade Gateway, jisql, Ribo)	Includes: • The Cascade Gateway – a gateway that acts as a proxy to provide a path to the database server if it is running on a different host from the Web server. • jisql – a graphical Transact-SQL editor written in Java that replaces SQL Advantage. • Ribo – a utility that captures, translates, and displays the Tabular Data Stream (TDS)™ protocol flowing between a TDS client and a TDS server.
jConnect Documentation	Contains the *Sybase jConnect for JDBC Programmer's Reference*.
Sybase Central ASE plug-in Java edition	The Java-based, graphical administration utility for Adaptive Server.
Sybase Central ASE plug-in C++ version 12.0	SQL Remote and the Monitor Server GUI require the C++ version of this plug-in. **Warning!** Do not use the Adaptive Server C++ plug-in for general administrative purposes.
Sybase ODBC drivers version 3.5	ODBC drivers have support for extended limits.
PowerDynamo version 3.5.2	A suite of tools for building and managing database-hosted Web sites with dynamic content.

Product and platforms	Description
InfoMaker version 7.0.3	A tool for personal data access, management, and reporting, for developers and end users. InfoMaker® allows you to create presentation-quality reports and powerful queries without the complexities of programming. It complements client/server business applications and development tools as well as desktop productivity suites.
SQL Remote version 7.0.2	SQL Remote™ enables two-way replication between a database server and multiple portable computer databases through e-mail or dial-up connections.
Physical Architect version 8.0	A tool for data modeling, including database design, generation, maintenance, reverse engineering, and documentation for database architects.

Sybase installation directory

The Sybase installation directory structure is created by the installation process. Adaptive Server is placed in the directory you indicate when you run the Studio Installer. Table 1-4 illustrates the installation directories that you are likely to access during the installation, configuration and administration of the Adaptive Server. This table is not comprehensive.

The Adaptive Server installation directory contains executable files and administrative tools, which are added as each product is installed.

Table 1-4: Installation directory for $SYBASE

Component directory	Subdirectory	Subdirectory
/ASE-12_5		
	/bin	
	/certificates	
	/debugger	
	/diag	
	/include	
	/init	/auditinit
		/bsrv
		/logs
		/sqlsrv
	/install	/spr
	/lib	
	/sample	/esp
		/JavaSql
		/server
	/scripts	
	/sybhelp	
	/upgrade	
	/xappdefaults	
/ASEP-1_0		
/CFG-1_0		
/charsets		
/collate	/unicode	
/configed		
/docs-45_55		
/EFTS-12_5		
/EJB-12_5		
/HOST-1_0		
/installed		
/Installer	/bin	
	/lib	

Component directory	Subdirectory	Subdirectory
/jConnect-4_5	*/classes*	*/com* */gateway* */sample*
	/devclasses	*/com*
	/docs	*/en*
	/gateway	
	/sample	
	/sp	
	/tools	
	/devclasses	
/jConnect-5_5	*/classes*	*/gateway2* */sample2*
	/devclasses	
	/docs	*/en*
	/gateway2	
	/sample2	
	/sp	
	/tools	
/jutils-2_0	*/cascade*	*/uk*
	/jisql	*/doc* */HelpFiles*
	/ribo	*/doc*
/locales	*/us_english*	
	/message	
	/unicode	
/OCS-12_5	*/bin*	
	/config	
	/devlib	
	/include	
	/lib	
	/lib3p (*/lib3p64* for 64-bit versions)	
	/locales	
	/sample	
	/sybhelp	
/shared-1_0		

Component directory	Subdirectory	Subdirectory
/SQLRemote	*/bin*	
	/scripts	
/Sybcent32		
/SYSAM-1_0	*/bin*	
	/licenses	

Installation Requirements

This chapter covers important information regarding system requirements for installing Adaptive Server databases and devices.

Topic	Page
System requirements	13
Product disk space requirements	16
Adaptive Server specifications	18
Adaptive Server devices and system databases	22

For easier installation, make a copy of this chapter, so you have the requirements information available as you perform the pre-installation, installation, or upgrade procedures.

System requirements

Table 2-1 shows the required version-level, RAM, and network-protocol requirements for Adaptive Server. See the release bulletin for the latest information on required system patches.

Table 2-1: Server system requirements

Hardware	Operating system	Supported protocols
HP9000/700, or HP9000/800, or 64-bit HP9000 D, K, V Series (The PA8000 chip is required.)	HP-UX 11.0, or HP-UX 11.0 running 64-bit runtime kernel	SPX or TCP
HP 9000/800 32-bit	HP-UX 11.0, or HP-UX 11.0 running 32-bit runtime kernel	SPX or TCP

Table 2-2: PC-client system requirements

Hardware	Operating system	Supported protocols
Windows NT	Windows NT 4.0, (Service Pack #6a Normal Server, or later)	TCP IPX/SPX Named Pipes
Windows 98	Windows 98	TCP IPX/SPX Named Pipes

Table 2-3 shows the system requirements for installing jConnect for JDBC.

Table 2-3: System requirements for jConnect on UNIX

Hardware	Any UNIX platform that supports Java VM 1.1.x
Operating system	Linux/Intel, Sun Solaris, HP UX, IBM AIX, DEC Digital Unix, SGI IRIX, and so on.
Java Developer's Kit	Sun's JDK 1.2.x or later
Database	• Adaptive Server Enterprise, from SQL Server 11.x through Adaptive Server Enterprise 12.x; or • Sybase SQL Anywhere 5.5.23 or later, running Open Server Gateway™ dbos50; or • Sybase SQL Anywhere Studio 6.0* **Note** Some features in jConnect version 4.5 and 5.5 require Adaptive Server Enterprise 12.x or later: • Support of wide tables • Support for outer join syntax • Support for Sybase Failover in high availability systems ("HA Failover") • Implementation of a DynamicClassLoader (or Adaptive Server Anywhere 6.x and later) • Support for distributed transaction management —Java Transaction API (JTA), and Java Transaction Service (JTS) For more information on these features, see the *Sybase jConnect for JDBC Programmer's Reference*.
Web browser	Any browser that supports JDK 1.1.x or later, such as Netscape 4.0.x, Internet Explorer 4.0.x, or HotJava 1.x
Other Sybase products (optional)	Adaptive Server IQ, OmniConnect, DirectConnect
Web server (optional)	For jConnect's TDS-tunnelling servlet, any Web server that supports the HTTPS protocol and javax.servlet interfaces

> **Note** A free evaluation version of SQL Anywhere Studio 6.0 (includes
> Adaptive Server Anywhere 6.0) is available from Sybase at
> http://www.sybase.com/products/anywhere/sqlanyform.html.
> The Open Server Gateway and the required DLLs are no longer needed with
> Sybase SQL Anywhere Studio version 6.0 and later.

Client products are supported on Windows NT, and Windows 98. Table 2-4
lists the RAM required for client products.

Table 2-4: RAM requirements for PC-client products

Product	Memory requirements
Adaptive Server plug-in for Sybase Central, Java Edition	16MB
Adaptive Server plug-in for Sybase Central, C++	16MB
Open Client/C Developer's kit	16MB, 32MB recommended
Monitor Client Library	16MB, 32MB recommended
Embedded SQL/COBOL, Embedded SQL/C	
jConnect 4.5 and 5.5	16MB each
SQL Remote 7.0.2	35MB
PowerDynamo 3.5.2	
InfoMaker 7.0.3	32MB
Physical Architect 8.0	16MB

Product disk space requirements

This section provides the approximate amount of disk space required by each
of the server components.

Table 2-5: Product disk space requirements (approximate)

Product and version level	Disk space required
Adaptive Server 12.5	191MB
jConnect 4.5	4.5MB
jConnect 5.5	4.5MB
jutilities (Cascade Gateway, jisql, Ribo)	6MB
jConnect documentation	6MB
Open Client/C 12.5	244MB
Language Modules (Server 12.5)	2–7MB per language
Language Modules (Open Client 12.5)	3–6MB per language
Monitor Server 12.5	3MB
Monitor Historical Server 12.5	6MB
Monitor Client Library 12.5	4MB
Enhanced Full-Text Search	70MB
EJB Server 32-bit platforms only	420MB

Language module sizes

Table 2-6 lists the sizes for the Adaptive Server and Open Client Language modules.

Table 2-6: Language module sizes (approximate)

Language	Size of Adaptive Server language module	Size of Open Client language module
Brazilian Portuguese	6.5MB	6.5MB
Chinese (simplified)	2MB	2MB
Chinese (traditional)	2MB	2MB
French	6.5MB	6.5MB
German	6.5MB	6.5MB
Japanese	4.5MB	4.5MB
Korean	2MB	2MB
Spanish	6.5MB	6.5MB

Adaptive Server specifications

This section provides system specifications for Adaptive Server on HP-UX.

Table 2-7: Adaptive Server specifications for HP-UX 32-bit processor

Hardware and memory		
Processor	32-bit	
Minimum RAM for Adaptive Server	64MB	32,768 2K pages
Number of shared memory segments	1 to 2	One segment for first GB; a second segment for next .75+ GB
Memory per user connection	Approx. 150K	With default stack size, packet size, and user log cache size
Default user stack size	45K	Minimum 30K

Table 2-8: Adaptive Server specifications for HP-UX 64-bit processor

Hardware and memory		
Processor	64-bit	
Minimum RAM for Adaptive Server	90MB	45,056 2K pages
Number of shared memory segments	1 to 2	One segment for first GB; a second segment for next .75+ GB
Memory per user connection	Approximately 190K	With default stack size, packet size, and user log cache size
Default user stack size	85K	Minimum 35K

Database specifications		
Databases per Adaptive Server	32,767	Practical limit is approximately 100
Maximum database size	2^{43} (8TB)	Minus overhead for system databases
Minimum allowable sybsystemprocs database	100MB	Required for an upgrade
Maximum size of a database device (disk partition)	2^{24} (32GB)	If the Operating System supports file sizes up to 32GB, then Adaptive Server supports file system devices up to 32GB
Maximum number of database devices per server	256	

Maximum number of devices or device pieces per database	Unlimited	
Maximum number of segments per database	31	
Maximum number of login IDs per server	2147516416	
Maximum number of users per database	2146484223	
Maximum number of groups per database	1032193	
pubs2 database	2MB	Minimum required for a new installation
	4MB	Free space required for an upgrade
pubs3 database	2MB	Minimum required for a new installation
	4MB	Free space required for an upgrade
interpubs database	2MB	Minimum required for a new installation
	4MB	Free space required for an upgrade
jpubs database	2MB	Minimum required for a new installation
	4MB	Free space required for an upgrade

Table specifications

User objects per database	$2^{31} - 100$	
Indexes per table	250 (one clustered index)	
Rows per table	Limited by available storage	Maximum 2^{32}
Columns per composite index	31	
Creation of clustered index	$1.2*(x + y)$ x = total data space in table, y = sum of space of all nonclustered indexes on table, and 20 percent overhead for logging	For sorted data, approximately 20 percent of the table size needed

Characters per database object name	30	

Query specifications

Maximum number of tables in a "union" query	256	
Maximum number of databases participating in one transaction	16	Includes database where transaction began, all databases changed during transaction, and tempdb, if it is used for results or worktables
Practical number of databases participating in one query	16	Includes each occurrence of each database queried and tempdb, if it is used for results or worktables
Maximum number of tables participating in a query	64	Maximum of 50 user tables, including result tables, tables referenced by views (the view itself is not counted) correlations and self-joins; maximum of 14 worktables
Maximum number of tables with referential integrity constraints for a query	192	

Procedure specifications

Number of buffers and procedure buffers	Configurable	Limited by amount of RAM and maximum size of shared memory segment
Minimum memory required per stored procedure	2K	
Maximum number of parameters per stored procedure	2048	

Adaptive Server 12.5 extended-limit capabilities vary by type of table and the database logical page size. Table 2-9 lists the column and row limits for allpages-locked (APL) tables.

Table 2-9: Allpages-locked (APL) tables

Maximum APL table limits	Number of columns	Column size 2K page	Column size 4K page	Column size 8K page	Column size 16K page
Fixed-length column	1024	1960 bytes	4008 bytes	8104 bytes	16296 bytes
Variable-length column	254	1948 bytes	3988 bytes	8068 bytes	16228 bytes

Table 2-10 lists the column and row limits for data-only-locked (DOL) tables.

Table 2-10: Data-only-locked (DOL) tables

Maximum DOL table limits	Number of columns	Column size 2K page	Column size 4K page	Column size 8K page	Column size 16K page
Fixed-length column	1024	1958 bytes	4006 bytes	8102 bytes	16294 bytes
Variable-length column	1024	1954 bytes	4002 bytes	8089 bytes	16290 bytes

Database space requirements depend upon the logical page size of the server. Table 2-11 lists the minimum size for each database.

Table 2-11: Database requirements for varying page sizes

Databases	2K page	4K page	8K page	16K page
Default database size	2MB	4MB	8MB	16MB
master database	6MB	12MB	24MB	48MB
model database	3MB	4MB	8MB	16MB
tempdb database	3MB	4MB	8MB	16MB

Larger logical page sizes can contain more data. Table 2-12 lists the maximum data for each logical page size.

Table 2-12: Data limits for tables according to page size

Tables	2K page	4K page	8K page	16K page
Maximum number of data bytes per text or image file	1800	3600	7650	16200
Bytes per index key	600	1250	2600	5300
User-visible row length DOL table	1958	4006	8102	16294
User-visible row length APL table	1960	4008	8104	16296

Adaptive Server devices and system databases

Devices are files or portions of a disk that are used to store databases and database objects. You can initialize devices, using raw disk partitions (for production systems) or operating system files (for nonproduction uses).

Adaptive Server requires the following devices:

- master device – to store system databases.

- sysprocsdev device – to store system procedures.

The master and sysprocsdev devices are created when you create a new Adaptive Server.

master device

The master device contains the following databases:

- master – controls the operation of Adaptive Server as a whole and stores information about all users, user databases, devices, objects, and system table entries. The master database is contained entirely on the master device and cannot be expanded onto any other device.

- model – provides a template for new user databases. The model database contains required system tables, which are copied into a new user database with the create database command.

- tempdb – the work area for Adaptive Server. Each time Adaptive Server is started, the tempdb database is cleared and rebuilt from the model database.

- The sample databases are stored on the master device at installation, but should be moved to a user-defined device after installation. For more information, see "Sample databases" on page 24 and "Optional devices and databases" on page 24.

Note For recovery purposes, Sybase recommends that you do not create other system or user databases or user objects on the master device.

sybsystemdb device and database

For new installations the master device also contains the sybsystemdb database. The sybsystemdb device stores the sybsystemdb database, which stores information about transactions in progress, and which is also used during recovery.

The sybsystemdb database is required to support distributed transaction management (DTM) features. Before installation, make sure you have enough space available on the default segment to support sybsystemdb.

See Chapter 8, "Upgrading Sybase Servers" for instructions on how to create the sybsystemdb device and sybsystemdb database.

Note For recovery purposes, Sybase recommends that you do not create other system or user databases or user objects on the master device.

sysprocsdev device

The sybprocsdev devices stores the sybsystemprocs database, which contains most of the Sybase-supplied system procedures. System procedures are a collection of SQL statements and flow-of-control statements that perform system tasks; for example, sp_configure.

The system procedures that are needed during recovery situations are stored in the master database.

Note sysprocsdev is the default system name for this device. However, it is frequently referred to as the sybsystemprocs device, since it stores the sybsystemprocs database.

Optional devices and databases

The following devices and databases are needed only if you configure Adaptive Server for optional functionality, like auditing or two-phase commit transactions.

sybsecurity device and database

The sybsecurity device stores the sybsecurity database and the auditing system procedures with which you can configure auditing for your system.

The auditing system records system security information in an Adaptive Server audit trail. You can use this audit trail to monitor the use of Adaptive Server or system resources.

Install auditing using the auditinit utility. The sybsecurity device is created as part of the auditing installation process.

See *Configuring Adaptive Server Enterprise* for instructions on how to configure Adaptive Server for auditing.

The auditing system is discussed in more detail in the *System Administration Guide*.

Sample databases

The pubs2 and pubs3 databases are sample databases provided as a learning tool for Adaptive Server. The pubs2 sample database is used for most of the examples in the Adaptive Server documentation; other examples use the pubs3 database. Both are available in U.S. English versions of Adaptive Server.

The interpubs database contain French and German data. jpubs contains Japanese data.

For information about installing the sample databases, see "Installing sample databases" on page 68.

For information on the contents of these sample databases, see the *Transact-SQL User's Guide*.

sybsyntax database

The syntax database, sybsyntax, contains syntax help for Transact-SQL commands, Sybase system procedures, Adaptive Server utilities, and Open Client routines. You can retrieve this information using the system procedure sp_syntax.

For example, to see the syntax of the Transact-SQL select command, enter:

```
sp_syntax "select"
```

Adaptive Server includes a script for creating the sybsyntax database.

For instructions on how to install sybsyntax, see "Installing online help for Transact-SQL syntax" on page 76.

For more information about sp_syntax, see the *Reference Manual*.

dbccdb database

The database consistency checker (dbcc) provides commands for checking the logical and physical consistency of a database. The dbccdb database stores the results of dbcc when dbcc checkstorage or dbcc check verifying are used.

dbcc checkstorage records configuration information for the **target database**, operation activity, and the results of the operation in the dbccdb database. Stored in the database are dbcc stored procedures for creating and maintaining dbccdb and for generating reports on the results of dbcc checkstorage operations.

For information on installing dbccdb see Chapter 25 of the *System Administration Guide,* "Checking Database Consistency."

dsync option *on* by default for database device files

The dsync option is used with the disk init and disk reinit commands. By default, Adaptive Server enables dsync for database device files. This option ensures that Adaptive Server can recover data from devices on file systems. However, dsync can cause a degradation in performance for device files that experience high write activity.

When you install a new 12.x Adaptive Server, by default dsync is set on for all devices.

When you upgrade a UNIX server that stores databases on UNIX file system devices, by default dsync is set:

- on for the master device
- off for all other devices.

See also, sp_deviceattr, and sp_helpdevice in the *Reference Manual*.

25

Determining the location, type, and size of a database device

Adaptive Server requires several database devices. Table 2-13 below shows baseline values for each of the devices. See the release bulletin for any last-minute changes to these values.

Table 2-13: Adaptive Server database devices

Device	Purpose	Minimum size	Minimum recommended size
master	Stores system databases	14MB for 2K pages 27MB for 4K pages 50MB for 8K pages 105MB for 16K pages	30MB 60MB 120MB 240MB
sysprocsdev (also called the sybsystemprocs device)	Stores the sybsystemprocs database	100MB	100MB (plus any space for holding stored procedures that you have created)
sybsystemdb	Transaction processing	4MB	5 – 20MB
sybsecurity (optional)	Required for auditing	5MB	7MB; more for specialized auditing

For all databases you can use either a raw partition or a file.

Sybase Adaptive Server supports the database devices on NFS- and CIFS-mounted devices with Network Appliance Filers for storing data. Network appliance filers provide the same performance and data integrity as raw devices. There are no changes needed to the operating system or to Sybase Adaptive Server to use network appliance filers.

NFS mounted devices have been tested on Solaris, HP/UX, IBM AIX, Windows NT, Linux, and SGI.

Note Previous versions of Adaptive Server performed buffered writes to file devices, which does not ensure recoverability. Starting in version 12.0, the default for writes to file systems is unbuffered, so files are now acceptable as database devices.

You may need to repartition the disk you choose. Contact your operating system administrator for assistance.

When preparing a raw partition device, follow these guidelines:

- Do not initialize a database device on the partition that contains your Sybase installation software. Doing so destroys all existing files on that partition.

- A raw partition designated for use by Sybase cannot be mounted for use by the operating system for any other purpose, such as for file systems or swap space.

- After a Sybase configuration utility or the disk init command has initialized a portion of a partition as a database device, the entire partition cannot be used for any other purpose. Any space left on the partition beyond the size specified for the device becomes inaccessible, unless you drop and re-create the device.

- For best performance, place the Sybase software and all disk devices, including the master device, on the same machine.

- To avoid any possibility of using a partition that contains the partition map, do not use cylinder 0.

- Place the database device on a character device, because the Adaptive Server recovery system needs unbuffered system I/O. The character devices are in the */dev* directory.

Note You can use either a block device or a character device. Unbuffered I/O is supported on a block device by using asynchronous I/O.

To determine whether a device is a block device or a character device, run this command on the */dev* directory:

```
ls -l
```

"b" (block) or "c" (character) appears in the left-most column.

❖ **Choosing a raw partition**

1 Determine which raw partitions are available.

2 Determine the sizes of the raw partitions.

3 From the list of available raw partitions, select a raw partition for each device, based on the size recommendations in Table 2-13 on page 26.

4 Verify with the operating System Administrator that the partition you have chosen is available.

5 Make sure the "sybase" user has read and write privileges to the raw partition.

Note For more information on choosing a raw partition, see your operating system documentation.

CHAPTER 3   **Sybase Software Asset Management (SySAM)**

This chapter describes licensing concepts that you need to know before you begin installing Adaptive Server.

Sybase recommends that you read through this chapter before beginning the installation of Adaptive Server in the enterprise environment.

Topic	Page
SySAM overview	29
Types of SySAM systems	34
SySAM administration	38
Adding feature licenses	39

This chapter describes important concepts regarding how to license optional Adaptive Server features, add new licenses to existing servers, set up the license manager in a network environment, and set up redundant servers for high availability and failover using Sybase Software Asset Management (SySAM).

You will need to refer back to this chapter when you begin installing Adaptive Server in a network environment.

SySAM overview

SySAM (Sybase Software Asset Manager) is a licensing mechanism that:

- Provides System Administrators with a means to monitor their site's use of Sybase products and optional features.

- Enables select Adaptive Server 12.5 features.

- Enables failover to other registered installations of Adaptive Server in the event of failure.

Adaptive Server features licensed through SySAM

You must register the Adaptive Server Enterprise license through SySAM before you can enable any of the optional features.

Table 3-1 describes the Adaptive Server features that are licensed through SySAM.

Table 3-1: Licensed Adaptive Server features

Feature name	License name	Description
Adaptive Server	ASE_SERVER	The basic Adaptive Server Enterprise product, without optional features.
High availability	ASE_HA	Adaptive Server failover capabilities for high availability environments.
Java in Adaptive Server	ASE_JAVA	Java and XML support in Adaptive Server databases.
Advanced security mechanisms	ASE_ASM	Network-based authentication and encryption using DCE and CyberSafe.
Distributed transaction management	ASE_DTM	Distributed transaction management support for XA and Microsoft DTC protocols.
Enterprise JavaBean Server	ASE_EJB	A transaction server provides the framework for creating, deploying, and managing middle-tier business logic in the form of EJBs in a multi-tier environment.
LDAP directory services	ASE_DIRS	Lightweight directory services.
Enhanced Full-Text Search	ASE_EFTS	Enhanced full-text search specialty data store.
External file system	ASE_XFS	Management of non-relational Web content.

If you plan to use Adaptive Server 12.5 with Replication Server, DirectConnect, or OpenSwitch, you must upgrade those products to the latest versions to ensure compatibility with new features from Adaptive Server 12.5.

Adaptive Server 12.5 features that are not yet supported with these complimentary products include:

• Extended row and column size limits

• Lightweight directory services (LDAP)

How SySAM works

SySAM "checks out" licensed features to users, and checks them back in when they are no longer needed. The basic components of SySAM are:

- One or more Adaptive Server features

- One or more license files

- The SySAM software, which consists of a license management daemon and a Sybase daemon

Figure 3-1 shows the relationship between these components.

Figure 3-1: SySAM components
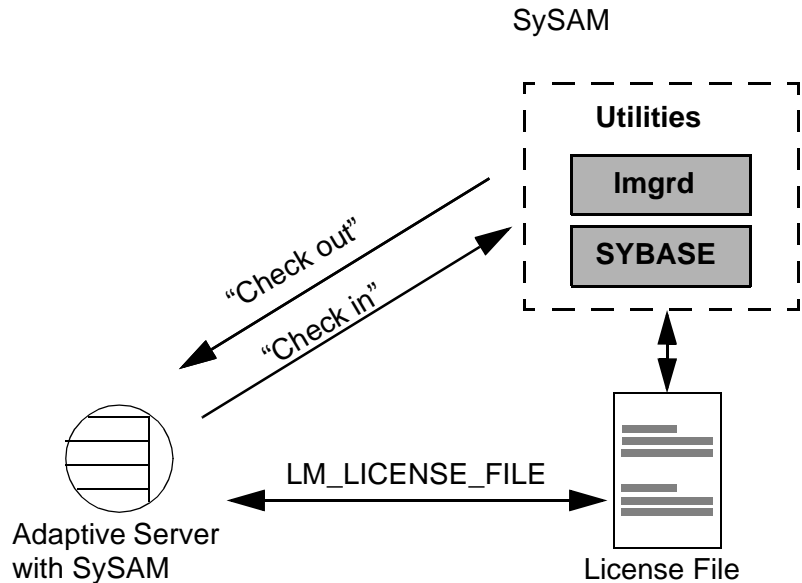
Adaptive Server check-out procedure

When you start Adaptive Server 12.5, it attempts to locate the license file *license.dat* located in *$SYBASE/$SYBASE_SYSAM/licenses*. If the file does not exist, Adaptive Server looks for the license file specified in the environment variable LM_LICENSE_FILE.

31

The license file contains a pointer to the SySAM software on the primary server known as the **license host**, or to a remote license host where SySAM is running. SySAM consists of a utility, lmutil, to manage licensing activities, and two daemons—the license management daemon, lmgrd, and the SYBASE daemon. The daemons handle requests to check in or check out licensed features, as shown in Figure 3-1.

Using information in the license file, Adaptive Server connects to SySAM and attempts to check out a license (ASE_SERVER) for the base Adaptive Server product. If the ASE_SERVER license is checked out successfully, Adaptive Server continues to start and attempts to check out any optional features enabled in *server_name.cfg*.

If Adaptive Server is configured to use optional features, such as distributed transaction management (ASE_DTM) or high availability (ASE_HA), it attempts to check out licenses for those features during the start-up process. If a license is not available for an optional feature, Adaptive Server still starts, but the feature cannot be used.

Starting Adaptive Server with optional features

You can use sp_configure parameters to turn on or off optional features. Table 3-2 lists the parameters used to configure optional features.

Table 3-2: Configuration parameters for optional features

Feature name	License name	Configuration parameter
High availability	ASE_HA	enable HA
Distributed transaction management	ASE_DTM	enable DTM
Java in Adaptive Server	ASE_JAVA	enable java
Advanced security mechanisms	ASE_ASM	use security services enable ssl
Enterprise JavaBean 32-bit platforms only	ASE_EJB	enable enterprise java beans
LDAP directory services	ASE_DIRS	n/a
Enhanced full-text search	ASE_FTS	enable full-text search
External file system	ASE_XFS	enable file access

Note Adaptive Server's support for distributed transaction management protocols: XA and Microsoft DTC protocols, requires configuration parameter enable DTM to be set and also a license to be available.

Distributed transaction management support through Adaptive Server Transaction Coordinator requires configuration parameter enable xact coordination to be set. However, this feature is available with no license requirements.

By default, the configuration parameters for optional features are set to 0 (off). To enable an optional feature, use sp_configure to set its configuration parameter to 1, and restart Adaptive Server.

Some features require additional preparation before you can use them. For more information, see Table 3-2 and *Configuring Adaptive Server Enterprise*.

- For information about configuring Adaptive Server as a companion server in a high availability system, see *Using Sybase Failover in a High Availability System*.

- For information about configuring Adaptive Server with distributed transaction, see *Using Adaptive Server Distributed Transaction Management Features*.

- For information about Java in Adaptive Server, see *Java in Adaptive Server Enterprise*.

Types of SySAM systems

The basic components of SySAM can be arranged for a standalone server, for a collection of servers on a network accessing a primary license host, and with multiple, redundant servers across a LAN or a WAN, for backup and load-balancing purposes.

Depending on which instance of Adaptive Server (primary or secondary servers) you are installing, the procedures for registering licenses vary.

Standalone system

In a standalone system, *all* of the SySAM components shown in Figure 3-2 reside on a single machine. The license file points directly to the asset management software, which runs on the same machine. This machine is called the **license host**.

Figure 3-2: Standalone system without optional features
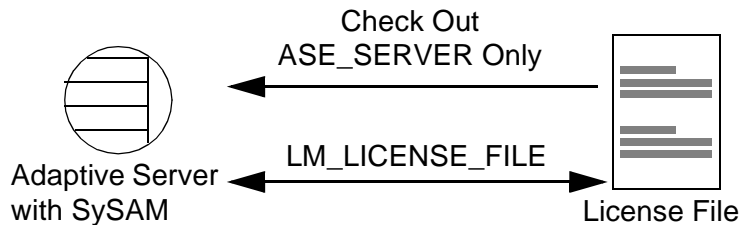
Network system

In a network system, the license host runs SySAM, which handles check-in and check-out requests from all **secondary servers**, as shown in Figure 3-3.

Figure 3-3: Network system
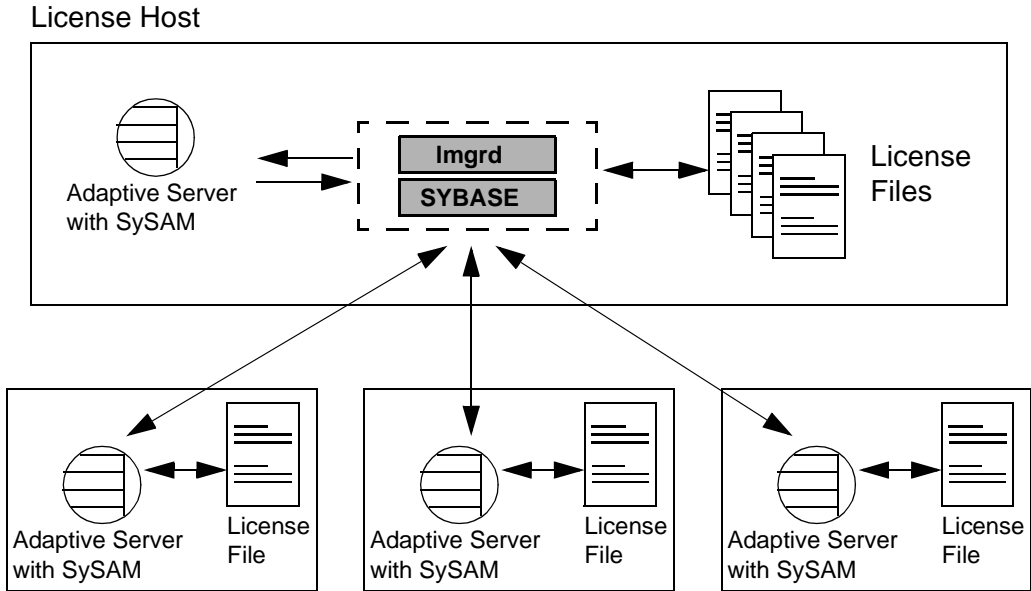
License Host

The license files for secondary servers contain only pointers to the license host. Secondary servers check out licenses from the license host through the SySAM software.

Note If you are using networked software asset management system, you must install the software on the license host before installing it on secondary servers.

Redundant servers

As part of a failover plan or a load-balancing system, you can have multiple servers running the same Adaptive Server configuration, using a redundant license across all servers. The **redundant server system** can be connected by a local-area network (LAN) or by a wide-area network (WAN).

A redundant server system is an excellent load-balancing mechanism for physically distant servers. The redundant license allows you to specify local servers as the first in the queue, and make remote servers available as backup. The SySAM application attempts to check out a license from a license-file list, starting with the first server. If that server fails for any reason, the second server in the list is contacted, and so on.

The redundant server system also provides failover protection. If one server in the list goes down for any reason, the second server responds to the license request. If that server fails too, the next server in the list is contacted, and so on.

SySAM in the network environment

Before you begin installing additional Adaptive Servers in a network system, the license manager must be installed and running on the license host. To verify that the license manager is running, see "SySAM administration" on page 38.

To install Adaptive Server on secondary servers, you need to have your Sybase License Certificate available. Also, you need to know:

* The host name of the license host.

* The port number on the license host where the asset manager listens for connections.

* The asset management software installation directory on the license host.

When installing Adaptive Server on secondary servers, the license manager prompts you for licenses after the installer has installed the products on to the hard drive.

1   The installer prompts: "Do you have Sybase Software Asset Management Certificates to register?"

    Click No.

2   The installer prompts: "Have you registered Sybase Software Asset Management Certificates at a central license host?"

    Click Yes.

3   Enter the following information about the license host from which this machine will be checking out Adaptive Server licenses:

    * License Manager Host is the host name of the license host on which you installed the asset management software.

    * License Manager Port is an unused port number on the License host. The local Adaptive Server uses this port number to contact the primary network node when checking out licenses.

4   Click OK.

    Continue with the installation procedures described in Chapter 4, "Installing Sybase Servers."

See Chapter 6, "Installing Sybase PC-Client Products" if you need help configuring client products.

Redundant servers

As part of a failover plan or a load-balancing system, you may have multiple servers running the same Adaptive Server configuration, using redundant licenses across all servers. The redundant servers can be connected by LAN or by WAN.

To configure redundant servers:

1   Follow the instructions in this guide to install Adaptive Server on each of the servers in the redundant system.

2   As you install the software, make a note of the machine names and the dedicated port number that SySAM uses on each machine.

> **Note** The Studio Installer automatically returns the machine name and a valid port number during the installation process.

3   For *each* machine in the redundant system, when the Studio Installer prompts for license information, enter:

- •   Order Number

- •   Feature Name

- •   Feature Count

- •   Software Version

- •   Authorization Code

The certificate information is provided in your Adaptive Server package on a printed Sybase Software Asset Management Certificate.

4   Click Continue Install.

Complete the installation of Adaptive Server. See Chapter 4, "Installing Sybase Servers."

5   After the installation is completed, and for each server in the redundant system, set the LM_LICENSE_FILE environment variable to:

```
"port@machine:port@machine:port@machine"
```

For example, if you use port 29722 on each machine and the machine names are Huey, Dewey, and Louie, the environment variable that points to the license file would look like this:

```
LM_LICENSE_FILE="29722@huey:29722@dewey:
29722@louie"
```

The first server in the license list is the first server queued by the licensing software.

SySAM administration

If you are installing servers in a network environment, SySAM must be installed and running on the license host before installing Adaptive Server on additional servers. This section describes how to start SySAM manually or as an automatic service, and other SySAM administration information.

Verify the software is running

To verify that the software is running:

When the lmgrd daemon starts, it automatically starts up the SYBASE daemon. To verify that the license management software is running on the system. Use:

```
$SYBASE/SYSAM-1_0/bin/lmutil lmstat -c
```

Make sure that both lmgrd and SYBASE are running before you continue the installation or start Adaptive Server.

If the SySAM software is not running, see "Starting the software manually" on page 38.

Starting the software manually

If the license manager is not running, you must start the license manager manually.

1   Use the UNIX command:

```
$SYBASE/$SYBASE_SYSAM/bin/lmgrd -c
$LM_LICENSE_FILE &
```

2   Alternatively, run the *startd.sh* script in the */bin* directory:

```
$SYBASE/$SYBASE_SYSAM/bin/startd.sh
$SYBASE/$SYBASE_SYSAM
```

Adding feature licenses

This section describes how to update the license file to enable Adaptive Server features.

To add features for which you have licenses:

1   Enable the new feature. See "Starting Adaptive Server with optional features" on page 32.

2   Shut down Adaptive Server. See Chapter 7, "Starting and Stopping Servers."

3   Verify that the license manager software is running. See "SySAM administration" on page 38.

4   Log on to the machine where the license manager is installed (license host).

5   Launch the license manager:

```
$SYBASE/$SYBASE_SYSAM/bin/lmgr
```

6   Click Yes when prompted: "Do you have Sybase Software Asset Management Certificates to register?" The SySAM License Manager screen prompts you for:

   •   Order Number

   •   Feature Name

   •   Feature Count

   •   Software Version

   •   Authorization Code

7   Click More until you have entered all licenses.

8   Click Done.

   Because you are adding licenses to an existing configuration, you must notify the license daemons of the changes by issuing a reread command.

9   Run the license management utility lmutil lmreread from the
    *$SYBASE/$SYSAM/bin/*directory.

```
$SYBASE/$SYBASE_SYSAM/bin/lmutil lmreread
```

The new license is appended to the end of the *license.dat* file.

If you encounter problems with new licenses, check the *lmgrd.log* file in
the *$SYBASE/$SYBASE_SYSAM/log* directory to see that there were
properly appended to the license file.

CHAPTER 4   **Installing Sybase Servers**

This chapter provides instructions for installing a new Adaptive Server, Backup Server, and Monitor Server.

Topic	Page
Overview	41
Installation definitions	42
Pre-installation tasks	42
Installing server components	46

Overview

The Studio Installer installs the Adaptive Server and related files on to the hard drive. The installation process allows you to install one or more of the following Sybase servers:

- Adaptive Server

- Backup Server

- Monitor Server

- EJB Server – 32-bit platforms only.

The basic process for installing Sybase servers is:

1   Perform the pre-installation tasks as described in "Pre-installation tasks" on page 42.

2   Perform the installation using any of the methods described in "Installation methods" on page 46.

3   Perform the post-installation tasks as described in Chapter 5, "Post-Installation Tasks."

Installation definitions

In this book, these terms are defined as follows:

- *Install* – to install the Adaptive Server files on to the hard drive.

- *Configure* – to configure attributes to make the Adaptive Server product fully functional.

- *Upgrade* – to configure an existing Adaptive Server to a more recent Adaptive Server version level.

- *Server* – provides a service in client/server computing. Examples include Adaptive Server, Backup Server, Historical Server, Monitor Server, and XP Server.

- *Client* – requests a service in client/server computing. Sybase Central, PowerDynamo, PowerDesigner, SQL Modeler, and end-user applications are clients.

Pre-installation tasks

Warning! If you install Adaptive Server for an upgrade, you cannot install it into the same directory as the currently installed version. The new version overwrites the old one, causing the upgrade to fail. You must have both the old and the new versions of Adaptive Server to perform an upgrade.

Table 4-1 is a list of the environment variables and their default settings.

Table 4-1: Environment variables

Variable	Set to...
$SYBASE	The installation directory. By default */usr/sybase*
$SYBASE_ASE	ASE-12_5
$SYBASE_EJB	EJB-12_5 Available on 32-bit platforms only.
$SYBASE_OCS	OCS-12_5
$SYBASE_SYSAM	SYSAM-1_0
$SYBASE_FTS	FTS-12_5

Variable	Set to...
$DISPLAY	*your_display*:0.0
Note You must manually set the $DISPLAY environment variable.	where *your_display* is the name of your machine. This variable must be set when performing a remote installation.

Before installing Sybase servers:

1   Read the release bulletins for the latest information on the products (Adaptive Server, Monitor Server, and so on) that you are installing. See the "Special Installation Instructions" section in the release bulletin.

2   Install operating system patches, if required. Required operating system patches are described in the release bulletin for the product.

Your Sybase product shipment includes printed release bulletins; they are also available at http://www.sybase.com/support/manuals.

3   Review the SySAM procedures and plan your client/server configuration using Chapter 3, "Sybase Software Asset Management (SySAM)."

4   Create a "sybase" account on your system to perform all installation tasks.

The "sybase" user must have permission privileges from the top (or root) of the disk partition or operating system directory down to the specific physical device or operating system file.

5   Log in to the machine as the "sybase" user.

It is important to maintain consistent ownership and privileges for all files and directories. A single user—the Sybase System Administrator with read, write, and execute permissions—should perform all installation, upgrade, and setup tasks.

6   Identify or create a directory location for the Sybase installation directory, where you install the server products.

If you are installing for an upgrade, see Chapter 8, "Upgrading Sybase Servers."

7   Verify that:

•   The directory location for the Sybase installation has sufficient space. Include an additional 2.4MB space for the installation program.

•   Your operating system meets the version-level, RAM, and network-protocol requirements.

- The installation directory does not have a space in the directory name. For example, installing Sybase products into the directory *Program Files* results in unpredictable behavior.

- The sybsystemdb database is required to support distributed transaction management (DTM) features. Before installation, make sure you have enough space available on the default segment to support sybsystemdb. See Chapter 2, "Installation Requirements."

Space requirements for Adaptive Server Enterprise products are listed in Chapter 2, "Installation Requirements."

8 If you are performing an installation on a remote machine, set the DISPLAY variable of the local X11 server.

- On the local machine where you are monitoring the installation of Adaptive Server, enter:

```
% setenv DISPLAY ip_address:0.0
```

where *ip_address* is the address of the machine on to which you are installing the software.

- On remote machine where you are installing Adaptive Server, enter:

```
$ xhost +
```

9 Verify that your network software is configured.

Sybase software uses network software even if Adaptive Server and Sybase client applications are installed on a machine that is not connected to a network.

If you are having connection problems, or if you want to verify your network configuration, complete these steps:

a Log in to your host machine with this UNIX command:

```
telnet host
```

This command gives you a login prompt for the same machine.

b Without logging out, log in to your host machine a second time, using the same command.

If you cannot log in to the machine a second time, your network software is not properly configured to run in loopback mode. See the network software documentation, or consult your vendor.

c Log out twice.

10 Adjust the operating system shared memory parameter.

For Adaptive Server to run, the operating system must be configured to allow allocation of a shared memory segment at least as large as the Adaptive Server total logical memory configuration parameter.

The default total logical memory parameter for HP is 32,768 2K pages (64MB) on the 32-bit operating system, and 45,056 2K pages (88MB) on the 64-bit operating system. To adjust the shared memory value of the operating system, use the System Administration Manager (SAM).

After you install Adaptive Server, you can change any configuration parameter, procedure cache, and data cache size. This may require the configuration parameter max memory to be increased.

11   Adjust shared memory segments.

Depending on the number and types of devices you use for backup (dump) and recovery (load), you may need to adjust the shared memory segment parameter in the operating system configuration file to accommodate concurrent Backup Server processes. The default number of shared memory segments available for process attachments is 6.

Adaptive Server may allocate shared memory segments after start-up if any reconfiguration through sp_configure requires additional memory. You may need to account for these additional segments. This can be avoided by allocating all the shared memory during the start-up, for the maximum memory you will make available to Adaptive Server, by using the allocate max shared memory configuration parameter. See the *System Administration Guide* for more information.

12   Enable asynchronous disk I/O.

This step is mandatory for HP users.

To improve I/O performance on character or raw block devices, enable asynchronous I/O by installing the HP asynchronous I/O driver from SAM. For help with installing this driver, contact your operating system administrator or HP technical support.

Note Before executing the following instructions, shut down Adaptive Server (or SQL Server).

To enable asynchronous I/O:

a   From the SAM Kernel Configuration menu, choose Drivers and set the Pending State for *asyncdisk* to In.

b   From the Actions menu, rebuild the kernel, and restart the system.

45

    c   At the UNIX prompt, execute the following statements as "root". The user ID of the user who is starting Adaptive Server and Backup Server must be the owner of the */dev/async* directory.

```
/etc/mknod /dev/async c 101 4
chmod 0660 /dev/async
chown sybase /dev/async
```

Note Setting the minor device number on */dev/async* to 4 reduces or eliminates transient 605 errors in Adaptive Server.

13 Determine the location, type, and size of each database device.

You must provide this information during the installation of Adaptive Server. Devices are operating system files or portions of a disk (called raw partitions) used to store databases and database objects. For details, see "Determining the location, type, and size of a database device" on page 26.

Installing server components

Follow the instructions in this section to install Adaptive Server, Backup Server, and Monitor Server.

Be sure that you are logged in as "sybase", and that you have performed the "Pre-installation tasks" on page 42.

Installation methods

Use any of the following methods to install Sybase servers:

•   Studio Installer – use the Studio Installer to install servers and customize them for a production environment. Fully customizing a server at installation time reduces the need to make changes later. Use this method to license your Adaptive Server optional features during the installation process.

- Resource file – first, use the Studio Installer to install the server components on to the hard drive, then run the srvbuildres utility to install and configure Adaptive Server and Backup Server for additional sites that require identical servers. The resource file also allows installation in environments in which X-Windows is not available. You cannot install Monitor Server and XP Server using resource files.

- You can access the installation utility, asecfg in the *$SYBASE/$SYBASE_ASE/bin* directory to configure a new server, upgrade your existing server, and change the default language of the server.

Installing components with Studio Installer

The Studio Installer creates the target directory (if necessary) and installs all the selected components into that directory.

At the end of the installation, you can verify the product installation. You may need to perform additional configuration procedures before you can use some products.

As part of the installation, the Studio Installer sets most of the environment variables needed for Adaptive Server products. However, you must source other environment variables by running the *SYBASE.csh* or *SYBASE.sh* script file after exiting the Studio Installer as described in Chapter 4, "Installing Sybase Servers."

Note Studio Installer does not set jConnect or Java utility environment variables—you must set them manually using the instructions in Chapter 5, "Post-Installation Tasks."

To install server components:

1   Insert the Server CD in the CD-ROM drive.

2   Log out, log on as "root", mount the CD, log out as "root", and log on again as "sybase", and start the Studio Installer.

```
cd /cdrom
./install
```

3   Select the type of installation to be performed.

- *Standard Install* – installs the default components a user needs.

- *Full Install* – installs every component on the CD.

- • *Customized Install* – allows you to select the components to install.
  Certain components are automatically installed if they are required to
  run other selected components.

  Backup Server, Monitor Server, and XP Server are, by default,
  installed with Adaptive Server.

The Studio Installer generates an error message and stops the installation
if you do not select any components for installation.

Note The Studio Installer automatically installs any components that are
required by the selected components, regardless of whether you deselected
them.

Click Back to select components, or Cancel to cancel the installation
procedure.

4   Click Next.

5   Enter the target directory and click Next to proceed.

   If you select Customized Install, the next window is the Component
   Selection screen, which allows you to specify which components to install.

   Components that would be installed in a standard installation appear with
   a check in the check box to the left of the product name. You can select or
   deselect components from this list. Components with subcomponents have
   a More... button enabled. Click this button to select or deselect
   subcomponents.

6   The Summary screen displays every component that is installed by the
   Studio Installer, the disk space required for each component, and the
   available disk space.

   If the target directory does not have enough free space, the available space
   appears in red. Clicking Next without sufficient hard disk space results in
   an error and stops the installation.

   Click Next.

7   If the target directory does not exist, the Studio Installer prompts you to
   create it.

   Click Yes to proceed.

Optionally, you can select Save from the Summary screen to save all the installation information into a *cmdfile* to proceed with the installation in a noninteractive, silent install. (See the Appendix, "Alternative Installation Methods.")

The Studio Installer installs the components on to the hard drive and displays a progress indicator.

Warning! Do not interrupt the installation process. If you do, you must manually remove all of the Adaptive Server and related files, and restart the installation in a clean environment.

Sybase Software
Asset Manager
(SySAM)

If you install any components without the appropriate license information, only Adaptive Server 12.5, without licensed features, is enabled.

Warning! The following SySAM license manager instructions are for the installation of Adaptive Server on the primary license host. If you are installing Adaptive Server on a secondary server, see the SySAM licensing instructions in "SySAM in the network environment" on page 36.

1   The Studio Installer prompts: "Do you have a Sybase Software Asset Management Certificate to register?"

Click Yes.

2   Enter information from the Sybase License Certificate for each Adaptive Server feature you have purchased. Entries are case sensitive.

   •   Order Number: Enter your Sybase order number.

   •   Feature Name: Enter the name of the Adaptive Server feature.

        Valid Adaptive Server feature names are:

        •   ASE_SERVER

        •   ASE_JAVA

        •   ASE_EJB

        •   ASE_HA

        •   ASE_DTM

        •   ASE_DIRS

        •   ASE_DIRS

- • ASE_XFS

- • Feature Count: Enter your license count number.

- • Software Version: Enter the Adaptive Server software version.

> **Warning!** You must enter the licenses exactly as they appear on the certificate.
>
> The license certificates for the base server and some of the options indicate version 12.0, although the software is at version 12.5. The server and all features are considered to belong to the same "Version 12 Product Family." The server and the options expect the license information to be entered exactly as it appears on the certificate.

- • Authorization Code: Enter the license key for the purchased feature.

The installer records the information for the current feature in the license file and prompts you to enter information for an additional feature.

See Chapter 3, "Sybase Software Asset Management (SySAM)" for detailed information about using the license manager.

3   Click More. . . if you have purchased additional licensed features. The installer records the information for the current feature in the license file and prompts you to enter information for an additional feature.

4   Click Continue Install after you have entered information from all of the Sybase License Certificates you have purchased. The installer records all license information and prompts you to configure the components you have installed.

If you encounter problems, check the installation log file to see a record of the installation process. The file is located in *$SYBASE/Installer.log*.

5   When the Studio Installer has completed the installation process, it asks if you want to configure the newly installed products and immediately launches the configuration utility if you select Yes.

Configuring the servers

The Studio Installer sets all the necessary environment variables to perform the configuration of Sybase servers. If your environment gets corrupted, the Studio Installer generates script files that you can run to correct the environment after the installation. If you need to correct the environment, run the following scripts as the "root" user.

*   If you are using a Bourne shell, run:

        $. SYBASE.sh

*   If you are using a C shell, run:

        % source SYBASE.csh

When you log in as "sybase" for the first time after installation, verify that the SYBASE environment variable is correctly set and that *$SYBASE/$SYBASE_ASE/bin* and *$SYBASE/$SYBASE_OCS/bin* are in your search path.

❖   **Configuring Adaptive Server**

1   The Studio Installer automatically begins the configuration process.

2   Enter the Sybase directory for this configuration. By default the installation directory displays.

    Click Next.

3   The Build or Upgrade Adaptive Server, Backup Server, and Monitor Server screen is displayed. From the drop-down list, select Build new servers.

    The srvbuild screen displays.

    Only the server types that were installed display on the srvbuild screen. For example, if you did not install Monitor Server, it does not appear on the srvbuild screen.

51

Figure 4-1: Server selection screen

4   Select, at a minimum, Adaptive Server. Click the check box to the left of the server name.

You can choose all server types at once or any combination of server types.

srvbuild consecutively displays the attributes screen for each server you choose, in the order listed on the screen. For example, if you chose Adaptive Server, Backup Server, and XP Server, but did not select Monitor Server, the attribute screen for Adaptive Server appears first. When you finish choosing the attributes for Adaptive Server, and click OK, the Backup Server attribute screen appears. When you are finished with that screen, the XP Server attribute screen appears. The Monitor Server attribute screen does not appear because you did not check the box for installing that product.

5   Click OK.

6   Provide a name for each server you install.

Note Do not change the name of your Backup Server.

---

srvbuild provides a default server name using the name of the machine you are installing on. You can change the default name.

Server naming conventions are as follows:

•   The name should not exceed 30 characters.

- The initial character must be a letter.

- The characters that follow must be letters, numbers, or underscores
  (_).

- Characters not allowed are: control characters, spaces, ampersands
  (&), asterisks (*), periods (.), and question marks (?).

- Use a name that is descriptive of your intended use and unique to your
  network.

- Use the extensions ("_back", "_mon", "_XP") to relate other server
  types to Adaptive Server.

- The XP Server name must be all uppercase, followed by an
  underscore (_) and the two-letters "XP". Although srvbuild does this
  for you automatically, you should be aware of this requirement in case
  you need to modify the XP Server name in the *interfaces* file.

All server names are stored in the *interfaces* file by srvbuild and are used
by clients and other servers to establish communication.

7   Click OK.

8   From the drop-down list at the top of the Adaptive Server attribute screen,
    select the logical page size for this installation of Adaptive Server. All
    pages in this installation will have the logical page size you specify here.
    If this installation of Adaptive Server has a logical page size of 2K, and
    you decide that you want a logical page size of, for example, 8K, you must
    create a separate installation of Adaptive Server with the 8K page size.

    The size of the master database is automatically allocated to accommodate
    the logical page size. You can increase the master database size if you
    want.

    If you are adding additional server languages for localization, you need an
    additional 2–7MB of space, per language in the master database. See
    Chapter 2, "Installation Requirements" for language module sizes.

    Note If you plan to use jdbc connections or the EJB Server, you must
    configure Adaptive Server to use iso_1 as the default character set.

    For production systems, Sybase recommends that you use a disk partition
    instead of an operating system file, for the master device. This example
    specifies a raw partition:

    ```
    /dev/rdsk/disk_partition
    ```

If you entered a disk partition as the master device, only the master device can occupy the disk partition. Set the size of the partition to the recommended 30MB.

If you entered an operating system file for the location of the master database, you can provide a size that is less than the default size, but the minimum allowable size is 25MB. The maximum size is the size of the file system, up to 2GB.

9 Provide an operating system file for the master and sybsystemprocs device paths:

```
/op_system_dir/master.dat
/op_system_dir/sybsystemprocs.dat
```

where *op_system_dir* is the name of the operating system directory you are going to use.

Note sysprocsdev is the default system name for this device. However, it is frequently referred to as the sybsystemprocs device, as in the Adaptive Server attribute screen, since it stores the sybsystemprocs database.

Sybase recommends that the sybsystemprocs device be located on a device separate from the master device. The sybsystemprocs database usually increases in size from version to version.

10 Provide an error log path name.

The default is:

$SYBASE/$SYBASE_ASE/install/errorlog

11 For Directory Services, specify the transport type, host name, and port number.

These entities define how servers and clients find and communicate with each other on the network. srvbuild adds this entry to the *interfaces* file during the installation process.

• Transport type – the menu provides a list of network protocols supported by Sybase. TCP is the default. Some network protocols in the menu may not be valid for your platform. Check with your operating system administrator.

• Host name – do not change the host name. The host name is the machine name where you are installing Adaptive Server.

- • Port number – if you did not change the default protocol, the port number displayed is the first available port address. The port number you specify cannot be used for any purpose other than running Adaptive Server.

  If you change the default port number, verify that the new port is not in use.

  To verify that a port number is available, use:

  ```
  netstat -a |grep port_number
  ```

  where *port_number* is the number you are checking. If this command does not return any results, the port number is not in use.

  If you selected a protocol other than the default, you may not be able to use the first available port number supplied. Check with the operating system administrator to see what port numbers are available.

Note If you want to use lightweight directory services for server connections instead of the *interfaces* file, see *Configuring Adaptive Server Enterprise* for your platform.

---

12 Click Edit Advanced Adaptive Server Attributes if you want to:

- • Specify a sybsystemdb device path for the sybsystemdb database. See the *Configuring Adaptive Server Enterprise*.

- • Place shared memory files in a directory other than the default location under *$SYBASE/$SYBASE_ASE*.

- • Change your default Backup Server.

13 To create the server, click Build Server. To continue configuring other Sybase servers, select OK.

srvbuild displays a Status Output screen that shows the status of the various installation tasks as they execute.

❖ **Configuring Backup Server**

This procedure assumes that you have completed the "Pre-installation tasks" on page 42.

1 The related Backup Server name is provided by default. It is the same name as the Adaptive Server with the extension "_back". This is the name of the Adaptive Server that will use this Backup Server.

2   Provide the Sybase System Administrator (SA) user name for the related Adaptive Server, or accept the default value.

    If the related Adaptive Server is created in the same srvbuild session as Backup Server, this field does not appear.

3   Provide the Adaptive Server SA password. This is the password for the related Adaptive Server.

    If the related Adaptive Server is created in the same srvbuild session as Backup Server, this field does not appear.

4   Provide an error log path for Backup Server.

5   Provide a location for the tape configuration file.

    Backup Server reads this file when a dump command is issued, to determine the device type and characteristics of the dump device specified in the dump command. Having access to this information speeds up the dump (recovery) process.

6   Specify the language you want Backup Server to use for its error messages. The default language is U.S. English.

    The language available depends on which language modules were installed on your system. By default, all character sets are copied from the CD when performing a standard or full install.

7   Specify the character set you want Backup Server to use.

    The default character set is Roman8.

 Warning! jConnect system stored procedures do not support the Roman8 character set. If you plan to use any jdbc connections or the EJB server, you must use iso_1 instead.

Other options are:

•   cp850: Code Page 850 (Multilingual) character set

•   cp437: Code Page 437 (United States) character set

•   mac: Macintosh default character set for Western European locales

•   ISO 8859-1 (Latin-1) Western European 8-bit character set

•   UTF-8 character set

8   Specify the maximum number of network connections.

This option specifies the total number of network connections available if you are backing up or restoring a database over the network to a remote Backup Server.

This option is not necessary if you are backing up to the default Backup Server, which is local. Set to 0.

9   Specify the maximum number of server connections. You can accept the default or change it.

This option specifies the number of server connections to Backup Server. Backup Server requires at least:

*   Two connections for each dump operation

*   One connection for each load operation

*   One connection for volume change messages

Allow a maximum of three times the number of expected concurrent dump and load sessions.

10 For the Directory Services, specify the transport type, host name, and port number.

These entities comprise the *interfaces* file entry. This entry defines how Backup Server and Adaptive Server find and communicate with each other on the network. srvbuild adds this entry to the *interfaces* file during the installation process.

*   Transport type – the menu provides a list of network protocols supported by Sybase. TCP is the default. Some network protocols on the menu may not be valid for your platform. Check with your operating system administrator.

*   Host name – the host name is the machine name where you are installing Backup Server.

*   Port number – if you did not change the default protocol, the port number displayed is the first available port address.

    The port number you specify cannot be used for any purpose other than Backup Server.

    If you change the default port number, verify that the new port is not already in use. Check with your operating system administrator.

    If you selected a protocol other than the default, you may not be able to use the first available port number. Check with the operating system administrator to see what port numbers are available.

11   Click OK.

❖   **Configuring Monitor Server**

This procedure assumes that you have completed the "Pre-installation tasks" on page 42.

1   If you selected Monitor Server to be installed, the Monitor Server attribute screen displays.

2   Provide the related Adaptive Server name. This field does not appear if the related Adaptive Server is created in the same srvbuild session as Monitor Server.

This is the name of the Adaptive Server that Monitor Server will be monitoring.

3   Specify the Sybase System Administrator (SA) user name. This field does not appear if the related Adaptive Server is created in the same srvbuild session as Monitor Server.

This option is required unless the USER environment variable is set.

This name specifies the Monitor Server superuser. It must be a valid login account for the Adaptive Server to be monitored. For production systems, Sybase recommends that you use the "sa" or "sybase" user account as the Monitor Server superuser.

Monitor Server uses the superuser account to connect to Adaptive Server to run special stored procedures and to verify shutdown requests that can be performed only by the superuser.

4   Provide the Adaptive Server SA password. This is the password of the Monitor Server superuser. This field does not appear if the related Adaptive Server is created in the same srvbuild session as Monitor Server.

If you omit the password, Monitor Server prompts for a password during start-up.

5   Determine the maximum number of connections allowed to Monitor Server. Valid values are 1–20.

A connection to Monitor Server can be any of:

•   A monitor in Sybase Central,

•   An active Historical Server session or a scheduled Historical Server recording session, or

- A connection to Monitor Server from a Monitor Client-Library application.

For example, if a user starts a Historical Server recording session, and another user opens the Performance Summary Monitor in Sybase Central, two connections to Monitor Server are used. If Monitor Server is configured for five connections, there are only three more connections available.

6   Specify the error log path.

This is the path name of the Monitor Server log file, where informational and error messages are logged.

The default location is the directory from which Monitor Server is started and the error log name is *servername_mon.log*, where *servername* is the name of the server.

7   Specify the path of the configuration file.

This is the path name of the Monitor Server configuration file. A default configuration file is not provided. This file is optional, but if specified, the values in the configuration file override the default runtime parameters that are otherwise used by Monitor Server.

8   Specify the shared memory directory, where the *servername.krg* shared memory file is located and *servername* is the name of the Adaptive Server.

If you omit this parameter, Monitor Server looks for the shared memory in the *$SYBASE/$SYBASE_ASE* directory.

If you did not change the default location while configuring the Adaptive Server, the default value is correct. Otherwise, use the value you specified for the server.

9   For the Directory Services, specify the transport type, host name, and port number. These entities comprise the *interfaces* file entry. This entry enables Monitor Server and Adaptive Server to find and communicate with each other on the network. srvbuild adds this entry to the *interfaces* file during the installation process.

- Transport type – the menu provides a list of network protocols supported by Sybase. TCP is the default. Some network protocols in the menu may not be valid for your platform. Check with your operating system administrator.

- Host name – the host name is the machine name where you installed Monitor Server.

• Port number – if you did not change the default protocol, the port number displayed is the first available port address.

The port number you specify cannot be used for any purpose other than installing the Monitor Server.

If you change the default port number, verify that the new port is not already in use.

To verify that a port number is available, use:

```
netstat -a |grep port_number
```

where *port_number* is the number you are checking. If this command does not return any results, the port number is not in use.

If you selected a protocol other than the default, you may not be able to use the first available port number. Check with the operating system administrator to see what port numbers are available.

10 Click OK.

❖ **Configuring XP Server**

If you selected XP Server to be installed, the XP Server Attribute Editor screen displays.

1 Provide the related Adaptive Server name, which is the name of the Adaptive Server that will be using XP Server. If the related Adaptive Server is created in the same srvbuild session as XP Server, this field does not appear.

> **Note** The XP server name must be in all capital letters, and must follow the format *ASE-NAME_XP*.

2 Provide the Sybase System Administrator (SA) user name for the related Adaptive Server, or accept the default value.

If the related Adaptive Server is created in the same srvbuild session as XP Server, this field does not appear.

3 Provide the related Adaptive Server SA password.

If the related Adaptive Server is created in the same srvbuild session as XP Server, this field does not appear.

4 For the Directory Services, accept the default values provided.

5 Click Build Servers.

srvbuild displays a Status Output screen that shows the status of the various installation tasks as they execute.

6   When the installation is complete, the Status Output screen displays "Done". Click Exit to return to the main window. Click Exit, again.

❖   **Configuring EJB Server**

This procedure assumes that you have completed the "Pre-installation tasks" on page 42.

If you installed the EJB Server, the configurator launches the Configure EJB Server screen.

You can modify the port number in the file, *$SYBASE/$SYBASE_EJB/Repository/Listener/EJBServer_iiops.props*.

1   Enter a valid port number, and click Next.

2   The utility displays the message: "Configuration of Installed Products is Complete." Click OK.

3   The utility displays the message: "Install Complete." Click OK.

Adaptive Server and related products have been successfully installed, and minimally configured for use. See "Installing sample databases" on page 68, to begin experimenting with your servers, or see the *System Administration Guide* at http://www.sybase.com/support/manuals for more advanced topics.

Viewing the installation status output screen

Use the information in this section each time you click the Build Server button after specifying the attributes for the server.

It takes several minutes to build the servers.

1   Review the Status Output screen for informational messages about the installation process.

If the installation is not successful, see Chapter 10, "Troubleshooting."

If the installation is successful, you see a message similar to this on the Status Output screen:

```
Server server_name was successfully created.
 Done.
```

Adaptive Server	You have a running server, as well as various system databases and system tables. Some references that may help you get started using your new Adaptive Server are:

- "Initializing Database Devices" and "Creating User Databases" in the *System Administration Guide* for information about creating an Adaptive Server user database and its devices

- *Transact-SQL User's Guide* to learn how to write queries

Backup Server	For information on developing a backup and recovery plan and backing up and restoring user databases, see the *System Administration Guide*.
Monitor Server	Monitor Server requires some additional configuration after installation. See the *Adaptive Server Enterprise Monitor Server User's Guide*.
XP Server	For information on using extended stored procedures, see the *Transact-SQL User's Guide*.

CHAPTER 5   **Post-Installation Tasks**

After the installation process, you have a minimally configured Adaptive Server. This chapter describes post-installation tasks to administer Adaptive Server and install stored procedures and sample databases. For more information see *Configuring Adaptive Server Enterprise* for your platform.

Environment variables

It is crucial to the operation of Sybase products that the environment variables are set correctly. The Studio Installer sets environment variables during the installation process. During the installation process, the Studio Installer also generates a file that you can use to reset most of the environment variables. The list in this section describes Sybase environment variables.

Environment variables are set in the user's environment either interactively or by including them in the user's *.login* and *.cshrc* files (for C shell) or *.profile* file (for Bourne shell).

You can set environment variables by running the *SYBASE.csh* or *SYBASE.sh* script file after exiting the Studio Installer.

1   From a Bourne, enter:

```
    . SYBASE.sh
```

2   From a C shell, enter:

```
    source SYBASE.csh
```

The script file does not set jConnect or Java utility environment variables. See the product documentation at http://www.sybase.com/support/manuals.

The environment variables are:

•   DSLISTEN – you can set this to the name that Adaptive Server uses to listen for client connections when one is not provided on the command line during start-up. If you do not set DSLISTEN, and you provide no name during start-up, the Adaptive Server name defaults to the server name given at installation. You cannot reset DSLISTEN by running the SYBASE scripts.

•   DSQUERY – defines the Adaptive Server name that client programs try to connect to if an Adaptive Server is not specified by name during a connection attempt. If you do not set DSQUERY, and you do not supply the Adaptive Server name, clients attempt to connect to the server name provided during installation. You cannot reset DSQUERY by running the SYBASE scripts.

•   SYBASE – defines the path of the Sybase installation directory. The installation program sets up the variable SYBASE to point to the release directory specified during installation.

•   SYBASE_ASE – defines the installation directory of Adaptive Server components.

•   SYBASE_OCS – defines the installation directory of Open Client.

•   SYBASE_EJB – defines the installation directory to the EJB Server. The EJB Server requires a license, and is installed only if you selected the "full" installation option.

Note You must set this environment variable manually. The Studio Installer does not set this environment variable during the installation process.

•   SYBASE_SYSAM – points to the license-management software directory. You cannot rest SYBASE_SYSAM by running the SYBASE scripts.

- LM_LICENSE_FILE – optionally points to the *license.dat* file in:

  $SYBASE/$SYBASE_SYSAM/licenses

- PATH – specifies which directory path to search for executables.The Sybase executables are located in the installed component's */bin* directory.

  To use Java-based features such as XML, your PATH must specify the JDK 1.2 directory. For example: */opt/sybase/java1.2/bin*, where */opt/sybase* is the installation directory.

  When using the setup files: *SYBASE.csh* or *SYBASE.sh*, the following paths are added to the beginning of the PATH environment variable:

  $SYBASE/$SYBASE_ASE/bin:$SYBASE/$SYBASE_OCS/bin

- SHLIB_PATH – specifies which directory to search for executables. The Sybase executables are located in the installed component's */bin* directory.

  When using the setup files: *SYBASE.csh* or *SYBASE.sh*, the following paths are added to the beginning of the SHLIB_PATH environment variable:

  $SYBASE/$SYBASE_ASE/lib:$SYBASE/$SYBASE_OCS/lib:$SYBASE /$SYBASE_FTS/lib, etc.

Note For instructions on setting jConnect environment variables, see the documentation for jConnect.

Setting environment variables

During the installation process, the Studio Installer generates C-shell and Bourne-shell files containing environment variables. The files are located in:

$SYBASE/SYBASE.sh, *$SYBASE/SYBASE.csh*

If you install multiple versions of Adaptive Server, or if you intend to access different Adaptive Servers on your network, you may want to change the environment variables PATH, DSLISTEN, DSQUERY, SYBASE_ASE, SYBASE_OCS, and SYBASE.

Setting EJB server environment

To enable the EJB Server, you must:

- Set the EJB environment variable:

      setenv SYBASE_EJB EJB-12_5

- Edit the *exporttool.sh* and *deploytool.sh* files located in *$SYBASE/sybcent32*, so that the JDK_LATEST environment variable points to the location of your JDK 1.2.2.

  For example:

      JDK_LATEST=$SYBASE/$SYBASE_EJB/jdk/jdk1.2.2

Verifying that servers are running

To determine whether the servers you installed are running, do either of the following:

1   For each server, enter the following command at the UNIX prompt enter:

        $SYBASE/$SYBASE_OCS/bin/isql -Usa -P -Sserver_name

   where *server_name* is the name of the server you want to check. If the server is running, you will see the isql prompt:

        1>

2   At the UNIX prompt, enter:

        $SYBASE/$SYBASE_ASE/install/showserver

   showserver shows all Adaptive Server-related processes are on the system.

Verifying that you can connect to servers

Use isql or dsedit to perform a quick test. Use isql to connect to servers:

1   At the command prompt, enter:

        isql -Usa -Ppassword -Sserver_name

   where *server_name* is the Adaptive Server, Monitor Server, or Historical Server name.

   The command prompt is displayed if the login is successful.

2   To display the Adaptive Server version number, enter:

```
1> select @@version
2> go
```

Adaptive Server's version number is displayed. The output should show Adaptive Server at version 12.5.

Connecting to Adaptive Server via Sybase Central

Sybase Central provides a graphical user interface where you can perform Adaptive Server administrative tasks. The Sybase Central interface is very similar to the Windows Explorer, with a hierarchical list of all servers in the left pane and a details list of the selected server in the right pane. To select a server, click on it with the mouse in the left pane.

1   From a UNIX command line, enter:

```
source $SYBASE/SYBASE.csh
```

2   Then enter:

```
$SYBASE/sybcent32/scjview
```

3   Select Tools | Connect, then select the server to which you want to connect from the drop-down list.

4   At the Login window, enter, the System Administrator's user name and password.

You must log in as the System Administrator to perform administrative tasks.

5   Click OK.

Warning! The first time you log in to Adaptive Server, use the default "sa" user name and leave the password blank. After you log in for the first time, you should change the System Administrator's password. See "Setting the System Administrator password" on page 68.

The Adaptive Server plug-in to Sybase Central does not display all servers listed in the *interfaces* file. Instead, Sybase Central lists only those servers that you connected to earlier, or those servers that are started as Windows NT services.

To access a new server for the first time, select Tools | Connect to select a server listed in the *interfaces* file.

Sybase Central has online help files. To access the online help, right-click on the server icon, the select Online help from the drop-down list. To access online help for Sybase Central, select Help from the menu bar, then select Sybase Central Help.

Setting the System Administrator password

A user account called "sa" is created for the Sybase System Administrator when you install the Sybase software. A user logged in as "sa" can use any database on Adaptive Server, including master, with full privileges.

Immediately after a new installation, there is no password on the "sa" account. The initial default value for the password is NULL. In a production environment, the Sybase System Administrator should always use a non-default password.

The Sybase System Administrator should log in to the new Adaptive Server as "sa" and set a password using sp_password:

```
$SYBASE/$SYBASE_OCS/bin/isql -Usa -P -Sserver_name
1> sp_password null, new_password
2> go
```

where null is the default password and *new_password* is the password that you are assigning to the "sa" account.

Installing sample databases

This section describes how to install the U.S. English and international language sample databases. For installation instructions specific to each sample database, see the following sections:

- "Running the database scripts" on page 70

- "Installing the interpubs database" on page 71

- "Installing the jpubs database" on page 71

The sample databases contain information about a fictitious business. You can use this information to learn about the Sybase products, without affecting essential data while learning.

Table 5-1 lists the scripts that you can use to install the sample databases.

Table 5-1: Sample database scripts

Script	Description
installpubs2	Installs the pubs2 sample database.
	This database contains data that represents a publishing operation. Use this database to test your server connections and to learn Transact-SQL. Most of the examples in the Adaptive Server documentation query the pubs2 database.
installpubs3	Installs the pubs3 sample database.
	This updated version of pubs2 uses referential integrity. In addition, its tables are slightly different than the tables used in pubs2. Where noted, the Adaptive Server documentation uses the pubs3 database in its examples.
installpix2	Installs the image data that is used with the pubs2 database.
	Note The master device size should be at least 30MB to install the full pubs2 database, including the image data. Run the *installpix2* script after you run *installpubs2*.
psdemodb	Installs the Powersoft sample database for PC-clients.
	For information on installing psdemodb, see *Configuring Adaptive Server Enterprise* for your platform.

Default devices for sample databases

The *$SYBASE/$SYBASE_ASE/scripts* directory contains scripts for installing the us_english sample database, foreign language sample databases, and the image data associated with the U.S. English pubs2 sample database.

If you have not used sp_diskdefault to change the status of the master device or to specify another default device, the scripts install the sample databases on the master device. Sybase does not recommend this configuration because it uses valuable space that is best used for system tables. Each sample database requires 3MB on your database device.

To avoid installing sample databases on the master device, either:

• Use sp_diskdefault to specify a default device other than the master device. For information on sp_diskdefault, see the *Reference Manual*.

- Modify each sample database installation script to specify a different device.

❖ **Running the database scripts**

1 Start Adaptive Server.

2 Determine the type (raw partition, logical volume, operating system file, and so on) and location of the device where you will be storing the pubs2 and pubs3 databases. You will need to provide this information later.

3 Make a copy of the original *instpubs2* and *instpubs3* scripts. Be sure you can access the copies, in case you have problems with the edited scripts.

4 Use a text editor to edit the script, if necessary, to specify a default device other than the master device, or use sp_diskdefault.

   From the Adaptive Server scripts directory (*$SYBASE/$SYBASE_ASE/scripts*), use isql to log in to Adaptive Server and run the script:

   ```
   isql -Usa -P***** -Sserver_name -iscript_name
   ```

   where *server_name* represents the destination server for the database and *script_name* is the full path to and file name of the script to run.

   For example, to install pubs2 on a server named VIOLIN, enter:

   ```
   isql -Usa -P***** -SVIOLIN
   -i $SYBASE/$SYBASE_ASE/scripts/installpubs2
   ```

5 To install the image data associated with pubs2 (pubs3 does not use image data), run:

   ```
   $SYBASE/$SYBASE_ASE/ isql -Usa -Ppassword
   -Sservername -i
   $SYBASE/$SYBASE_ASE/scripts/installpix2
   ```

Note The image data requires a fair amount of space—there are six pictures, two each in the PICT, TIFF, and Sun raster file formats. Run *installpix2* script only if you want to use or test the image datatype. Sybase does not supply any tools for displaying image data. You must use appropriate screen graphics tools to display the images after you have extracted them from the database.

For more information about running these scripts, see *Configuring Adaptive Server Enterprise*.

interpubs database

interpubs is a database similar to pubs2 that contains French and German data. This data contains 8-bit characters and is available for use at Adaptive Server installations using the ISO 8859-1 (iso_1) or Roman8 (for HP-UX) character set. To display the French and German data correctly, you must set up your terminal to display 8-bit characters.

❖ **Installing the *interpubs* database**

1   Be sure iso_1 or Roman8 is installed as the default character set or as an additional character set.

2   Determine the type (raw partition, logical volume, operating system file, and so on) and location of the device where you will be storing the interpubs database. You will need to provide this information later.

3   Make a copy of the original *installintpubs* script. Be sure you can access this copy, in case you experience problems with the edited script.

4   Use a text editor to edit the script, if necessary, to specify a default device other than the master device, or use sp_diskdefault.

5   Execute the script, using the -J flag to ensure that the database is installed with the correct character set:

```
isql –Usa –Ppassword –Sservername –Jroman8 -i
$SYBASE/$SYBASE_ASE/scripts/roman8/installintpubs
```

For more information on the -J option in isql, see the *Utility Guide*.

jpubs database

If you installed the Japanese Language Module with your Adaptive Server, the scripts file contains the *installjpubs* script for installing the jpubs database. jpubs is a database similar to pubs2 that contains Japanese data. *installjpubs* uses either the EUC-JIS (eucjis) or the Shift-JIS (sjis) character set.

Verify that you are running HP Native Language I/O to ensure that your system can display Japanese data correctly.

❖ **Installing the *jpubs* database**

To install the jpubs database:

1   Set your terminal to display 8-bit characters.

2 Verify that either the EUC-JIS or Shift-JIS character set is installed as Adaptive Server's default character set or as an additional character set.

3 Determine the type (raw partition, logical volume, operating system file, and so on) and location of the device where you will be storing the jpubs database. You will need to provide this information later.

4 Make a copy of the original *installjpubs* script. Be sure that you can access this copy, in case you experience problems with the edited script.

5 Use a text editor to edit the script, if necessary, to specify a default device other than the master device, or use sp_diskdefault. See "Default devices for sample databases" on page 69.

6 Execute the *installjpubs* script, using the -J flag to ensure that the database is installed with the correct character set:

```
isql -Usa -Ppassword -Sservername -Jeucjis
 -i $SYBASE/$SYBASE_ASE/scripts/eucjis/installjpubs
```

or:

```
isql -Usa -Ppassword -Sservername -Jsjis
  -i $SYBASE/$SYBASE_ASE/scripts/sjis/installjpubs
```

For more information on the -J option in isql, see the *Utility Guide*.

Maintaining the sample databases

The sample databases contain a "guest" user that allows access to the database by any authorized Adaptive Server user. The "guest" user has a wide range of privileges, including permissions to select, insert, update, and delete user tables. For more information about the "guest" user and a list of "guest" permissions, see the *System Administration Guide*.

If possible, and if space allows, give each new user a clean copy of the sample databases so that she or he is not confused by other users' changes.

If space is a problem, you can instruct the user to issue the begin transaction command before updating a sample database. After the user has finished updating one of the sample databases, he or she can issue the rollback transaction command to undo the changes.

jConnect 4.5, 5.5, and Java utilities

Before you install jConnect version 4.5, 5.5, jisql, or Ribo, you must install the Java Development Kit (JDK). JDK is installed by default when you perform a full or standard install with the Studio Installer.

When you install jConnect 5.5, you receive classes that are compatible with JDBC 2.0, regardless of which JDK (1.1.x or 1.2) version you use. The following table lists the JDK and jConnect versions necessary to develop applications that are compliant with different versions of JDBC.

If you are developing	Use JDK version	Use jConnect version
JDBC 1.x applications/servlets	1.1.x or compatible browser	4.x
JDBC 2.x applications/servlets	1.2.x or compatible browser	5.x

For information on developing applets with jConnect, see *jConnect for JDBC x.x Product Documentation* at http://www.sybase.com/support/manuals.

Note The Java VM security manager does not allow a version of jConnect installed in your local file system to create a connection to a server, even if the server is your local system. This can create a problem if you use Netscape to run an applet that uses jConnect.

When Netscape needs to load jConnect to create a connection to a server, it looks for a CLASSPATH defined in its environment to find the jConnect classes. If it finds a CLASSPATH that you have defined to point to your local jConnect installation, it loads jConnect directly from the local file system, rather than from the server, and tries to create a connection. This generates a security error and the connection fails.

Therefore, if you are using Netscape to run jConnect applets, do not set the CLASSPATH to the local jConnect directory.

Using open database connectivity

Some applications do not connect to Adaptive Server directly through the Open Client software but through the Open Database Connectivity (ODBC) driver, or the jConnect driver instead. For example, PowerDynamo connects through the ODBC driver.

ODBC and OLE DB require the Microsoft Data Access Component (MDAC).
You can download it directly from Microsoft's Web site at
http://www.microsoft.com/data.

See *Configuring Adaptive Server Enterprise* for your platform for more
information.

Installing stored procedures

By default, all Adaptive Server store procedures are installed during the
installation process. If you need to re-install the jConnect stored procedures for
any reason, see *jConnect for JDBC* documentation set at
http://www.sybase.com/support/manuals.

Installing character sets

By default, when Adaptive Server and Backup Server are installed on HP
systems, the installation installs the character set files for Roman8 which
supports the Western European languages.

Warning! If you plan to use Java features or jConnect, you must install the
iso_1 character set. Roman8 does not support Java.

In addition, the installation defaults include sort-order definition files for
Western European character sets and U.S. English system message files.

If you want Adaptive Server error messages to display in a language other than
U.S. English (us_english), you must install the appropriate language module.

Warning! Make all changes to the default character set and sort order for a new
Adaptive Server before creating any user databases or making any changes to
the Sybase-supplied databases. Changing the character set and sort order after
data or data structures have been added to Adaptive Server can cause incorrect
behavior. To change the character set or sort order after you have added data,
see the *System Administration Guide*.

A full install of Adaptive Server installs all the language components automatically. If you did not select a full install, you must install additional language modules manually.

❖ **Installing new language modules**

1   Install the language modules into the same directory where you installed Adaptive Server.

Note Each language uses about 2MB of database space per module. If necessary, use the alter database command to increase the size of the master database before adding another language. For more information, see the *System Administration Guide*.

---

2   Use charset to load the default character set and sort order.

Note Use charset only at the time of installation. For alternative methods, see the Localization chapter in *Configuring Adaptive Server Enterprise* for your platform.

---

3   To use charset, the server must be running and you must have System Administrator privileges. Use the *file name* of the sort order:

```
$SYBASE/SYBASE_ASE/bin/charset -Usa -Ppassword
-Sserver_name sort_order_file character_set
```

4   Replace *sort_order_file* with the name of the sort order file. Replace *character_set* with the Sybase name for your character set.

5   Use charset to load any additional character sets. See Table 5-2 for charset syntax.

Table 5-2: charset syntax

Syntax	`charset [sort_order_file	charset.loc]` `charset_directory`
Usage	`charset` `[ -U username ]` `[ -P password ]` `[ -S server ]` `[ -I interfaces ]` `[ -v version ]` `[sort_order_file	charset.loc]` `[charset_directory]`

❖ **Installing character-set definition files**

If you plan to use the Adaptive Server built-in character-set conversions, you must load the character-set definition files for all the character sets on your client platforms. If you are using the Unilib character-set conversions, you do not need to do this.

1   Using isql, log in to your server as "sa" and select the master database.

```
1> use master
2> go
```

2   Use the *ID* of the sort order to configure your server for the new character set and sort order.

```
1> sp_configure "default sort_order_id",
2> sort_order_id, "character_set"
3> go
```

3   Replace *sort_order_id* with the ID for your sort order. Replace *character_set* with the Sybase name for your character set.

4   Shut down the server to start the reconfiguration process.

Use the normal process on your UNIX system to restart the server, usually by invoking one of the *RUN_xxx* scripts from *$SYBASE/$SYBASE_ASE/install*.

5   The server starts, rebuilds all the system indexes, then shuts down. Restart a second time to bring the server up in a stable state.

To change the default character set and sort order of Adaptive Server after the initial installation, see the *System Administration Guide*.

Installing online help for Transact-SQL syntax

This section provides instructions for installing online help for Transact-SQL syntax.

Online syntax help: *sp_syntax*

There is a script for installing the syntax help database, sybsyntax. You can retrieve this data with the sp_syntax system procedure. The script is located in:

$SYBASE/$SYBASE_ASE/scripts/ins_syn_sql

For more information on sp_syntax, see the *Reference Manual*.

This script includes syntax information for Transact-SQL, the system procedures, and the Sybase utilities. When you execute this script, you install the SQL portion of the sybsyntax database.

When you first execute this script, it creates the sybsyntax database and the necessary tables and indexes. If you execute the script again, the previously installed rows of information are deleted from the table in the database and then reinstalled.

Default device for the *sybsyntax* database

By default, the sybsyntax installation scripts install the sybsyntax database on the device that is designated as the default database device.

Table 5-3 lists the minimum sybsyntax database requirements.

Table 5-3: sybsyntax requirements

Logical page size	2K	4K	8K	16K
Minimum database size	3MB	5MB	Same size as the model	Same size as the model

If you have not used sp_diskdefault to change the status of the master device (which is installed as the default disk) or to specify another default device, the scripts install sybsyntax on the master device. Sybase does not recommend this configuration because sybsyntax uses valuable space that is best left available for future expansion of the master database.

To avoid installing sybsyntax on the master device, do one of the following:

- Use sp_diskdefault to specify a default device other than the master device. For information about sp_diskdefault, see the *Reference Manual*.

- Modify each sybsyntax installation script that you plan to execute to specify a different device, as explained in the following section.

Installing *sybsyntax*

For each sybsyntax installation script you want to execute:

1   Determine the type (raw partition, logical volume, operating system file, and so on) and location of the device where you plan to store the sybsyntax database. You will need to provide this information later.

2   Make a copy of the original script. Be sure you can access this copy, in case you experience problems with the edited script.

3   Use a text editor to edit the script, if necessary, to change the default device from the master device to the device created in step 1. For information on the default device, see "Default device for the sybsyntax database" on page 77.

   •   Comment out the following section, which specifies the default device:

```
/* create the database, if it does not exist */
 if not exists (select name from sysdatabases
 where name = "sybsyntax")
 begin
     /* create the sybsyntax table if it doesn't exist */
     /* is the space left on the default database
     devices > size of model? */
     if (select sum (high-low +1) from sysdevices where status
     & 1 = 1) - (select sum(size) from sysusages, sysdevices
         where vstart >= sysdevices.low
         and vstart <= sysdevices.high
         and sysdevices.status &1 = 1) >
         (select sum(sysusages.size) from sysusages
         where dbid = 3)
     begin
         create database sybsyntax
     end
     else
     begin
         print "There is not enough room on the default
         devices to create the sybsyntax database."
     return
     end
 end
```

   •   After you have commented out this entire section, add this line to the script:

```
        create database sybsyntax on device_name
```

   where *device_name* is the name of the device where you want to install sybsyntax.

4   Execute the script:

```
isql -Usa -Ppassword -Sservername
    -i $SYBASE/$SYBASE_ASE/scripts/ins_syn_sql
```

where *sa* is the user ID of the System Administrator, *password* is the
System Administrator's password, and *servername* is the Adaptive Server
where you plan to install the database.

If you have set the DSQUERY environment variable to the *servername*,
you can replace the server name with DSQUERY. For example:

```
isql -Usa -Ppassword -S$DSQUERY
    -i $SYBASE/$SYBASE_ASE/scripts/ins_syn_sql
```

5   To ensure that you have installed the sybsyntax database and that it is
     working correctly, use isql to log in to the server on which you installed the
     database, and execute sp_syntax. For example:

```
isql -Usa -Ppassword -Sservername

1> sp_syntax "select"
2> go
```

Adaptive Server displays a list of commands that contain the word or word
fragment "select."

Configuring XP Server after initial installation

If you install Adaptive Server and XP Server in the same build session, srvbuild
automatically adds information about the XP Server to the sysservers table of
Adaptive Server. If you install XP Server in a different build session than the
Adaptive Server installation, you are prompted during the XP Server
installation process to supply the name of the related Adaptive Server and the
System Administrator's name and password. This information is required by
Adaptive Server to start XP Server.

If you do not supply this information during the installation process, XP Server
cannot run. When you execute an extended stored procedure (ESP), you see an
error message similar to the following:

```
Msg 11018, Level 16, State 1:
Procedure 'xp_cmdshell', Line 2
XP Server must be up for ESP to execute.
```

To add this required information to the sysservers table manually, enter:

```
sp_addserver servername_XP, NULL, servername_XP
```

79

where *servername* is the name of the Adaptive Server.

CHAPTER 6 **Installing Sybase PC-Client Products**

Your UNIX-based Adaptive Server also includes PC-client products that you install on a PC and connect, via the *interfaces* file, with your Adaptive Server.

This chapter describes how to install client products on Windows 98 and Windows NT 4.0.

Overview

Client plug-ins enable clients, such as Sybase Central and Open Client, to access Adaptive Server. See the *Open Client/Server Configuration Guide for Desktop Platforms*.

Sybase Central plug-In   The Sybase Central plug-in allows you to connect to Adaptive Server via client connections. You can use Sybase Central to connect to remote servers, local servers (installed on the same machine), and multiple servers. See "Configuring network connections for client products" on page 87. Before you use the plug-in, restart your computer to implement changes made to the PATH and CLASSPATH environment variables during installation.

Open Client runtime   The Sybase PC-client CD includes Open Client software for Windows NT and Windows 98. Several products, such as the Adaptive Server plug-in and SQL Advantage, require Open Client runtime. Monitor Client Library requires Open Client/C with the SDK option.

Open database   Some applications do not connect to Adaptive Server directly through the Open
connectivity   Client software but through the Open Database Connectivity (ODBC) driver, or the jConnect driver instead. For example, PowerDynamo connects through the ODBC driver.

See *Configuring Adaptive Server Enterprise* for more information.

ODBC and OLE DB require the Microsoft Data Access Component (MDAC). You can download it directly from Microsoft's Web site at http://www.microsoft.com/data.

Installing PC-client products

Before starting the installation process for PC-client products, be sure that you have at least an extra 2.4MB of disk space, a *\temp* directory and TEMP environment variable.

The installation program uses the extra space and directory to write files temporarily during the installation. The installation program frees this space after installation is complete.

Special instructions   Environment Space is an area of memory set aside for each process in which
for Windows 98   environment variables can be defined. By default, there is a fixed amount of
platforms: memory set aside to hold environment variables. This space contains the name of the variable (for example, TEMP) and its value (for example, *C:\Windows\temp*). Windows 98 by default provides for 256 bytes of environment space. Each environment variable defined takes up a specific amount of space depending upon the number of characters in the name and value. If you define an environment variable and there is not enough space to hold the variable and its definition, Windows will return an error indicating that you are "out of environment space."

If you are installing PC-client products on the Windows 98 platform, follow these steps to increase your environment space.

• Bring up a DOS window

• Right click on the MSDOS icon on the left end of the top window bar to bring up the properties window

- • Select Memory

- • Select Initial Environment and enter: 4096

- • Click OK

Warning! The *libtcl.cfg* file, located in *%SYBASE%\ini*, will be overwritten during the installation process.

To install client products on Windows 98 and Windows NT:

1   Verify that your computer has sufficient RAM for each product. See Chapter 2, "Installation Requirements."

2   If you are unloading components on Windows NT, log in using an account with Windows NT administrator privileges.

3   Close any open applications or utilities to free memory and system resources.

4   Insert the PC-client CD in to the drive. The Studio Installer should start automatically.

   Alternatively, you can start the Studio Installer from the Windows Start menu, select Start | Run, and enter:

```
X:\setup.exe
```

   where *X* is your CD-ROM drive.

5   Click OK.

6   When the Studio Installer starts, it prompts you for an installation directory. Click Next.

7   Select the type of installation to be performed.

   - • *Standard Install* – installs the default components a user needs.

   - • *Full Install* – installs every component on the CD.

   - • *Customized Install* – allows you to select the components to install. Certain components are automatically installed if they are required to run other selected components.

   Note If you select Customized Install, the next window is the Component Selection screen, which allows you to specify which components to install.

Components that would be installed in a standard installation appear with a check in the check box to the left of the product name. You may select or deselect components from this list. Components with subcomponents have a More... button enabled. Click this button to select or deselect subcomponents.

8   Click Next.

The Summary screen displays every component to be installed by the Studio Installer, the required disk space, and the available disk space.

If the target directory does not have enough free space, the available space appears in red. Clicking Next without sufficient hard disk space results in an error and stops the installation.

9   Click Next.

Note Optionally, you can select Save on the Summary screen to save all the installation information into a *cmdfile* to proceed with the installation in a noninteractive, silent install. See the Appendix, "Alternative Installation Methods."

10 If the target directory does not exist, Studio Installer prompts: "OK to create directory?" Click Yes.

The Studio Installer installs the components and displays a progress indicator.

Warning! If you are prompted to overwrite any DLLs, select Yes only if the version of the new DLLs is later (the version number is larger) than the one it is attempting to overwrite.

The Studio Installer generates an error message and stops the installation if you do not select any components for installation.

Click Back to select components, or Cancel to cancel the installation procedure.

11 When prompted to restart the computer, leave the CD in the CD-ROM drive and select Yes. Restarting the PC updates the environment variables and registry keys.

Note If you are installing client products on the Windows 98 platform, you must manually restart the computer before proceeding with the configuration of client products. The Studio Installer does not automatically restart the Windows 98 operating system.

To configure client network connections to Adaptive Server, see *Configuring Adaptive Server Enterprise* for your platform.

Note If you installed jConnect 4.5 or 5.5 from the PC-client CD, see "Installing Stored Procedures for jConnect" in Chapter 5, "Post-Installation Tasks."

Installing the Java Runtime Environment

Before you can use the Sybase Central monitoring features, you need to install the Java Runtime Environment (JRE):

1   Go to *%SYBASE%\ASEP_win32*.

2   Double-click the file *jdk1_1_8-win.exe* and follow the installation instructions.

3   Restart your computer after you install the JRE.

Installing the Java Development Kit

Optionally, you can install the entire Java Development Kit (JDK) instead of the JRE. The JDK requires more disk space and includes files and features that support development of Java applets and applications. Sybase does not provide the JDK (see step 1 in the following instructions).

1   Download the JDK for your operating system from the Java Software Web site: *http://java.sun.com/products/jdk/1.1*.

2   Double-click the file you downloaded.

3   Follow the installation instructions.

4   Add the following path to your CLASSPATH environment variable:

    `x:\`*`install_path`*`\lib\CLASSES.ZIP`

where *x:\install_path* is the location (drive and directory) where you installed the JDK; for example, *c:\javasoft.*

5   Add *x:\install_path\bin* to your PATH environment variable, where *x:\install_path* is the location (drive and directory) in which you installed the JDK; for example, *c:\javasoft*.

Installing InfoMaker

InfoMaker is distributed as part of the PC-client package. If you want to install InfoMaker and use the Demo Database, you must download and install Adaptive Server Anywhere.

To install InfoMaker:

1   Use the Studio Installer to install the software from the distribution media as described in Chapter 4, "Installing Sybase Servers."

2   Run the InfoMaker setup utility located in *%SYBASE%\INFO-7_0_2\install*.

From a DOS prompt, enter:

    `seteup.exe "-EAS3"`

Or, from the Windows task bar, select Start | Run, and enter the full path to the executable, including the -EAS3 argument. For example:

    `X:\sybase\INFO-7_0_2\install\setup.exe "-EAS3"`

where *X:\sybase* is the installation directory for the PC-client components.

3   When prompted for an installation directory, accept the default.

4   When prompted to restart the computer, leave the CD in the CD-ROM drive and select Yes. Restarting the PC updates environment variables and registry keys.

5   See the Powersoft documentation for instructions on using InfoMaker.

Note If you cannot launch the Demo Database through your shortcut, please change the properties in it by setting the path to your SQL Anywhere win32 directory.

Installing PowerDynamo

PowerDynamo is distributed as part of the PC-Client package.

To install PowerDynamo:

1   Use the Studio Installer to install the software from the CD as described in "Installing PC-client products" on page 82.

2   Run the InfoMaker setup utility located in *X:\sybase\PowerDynamo\install\setup.exe*

    where *X:\sybase* is the installation directory for the PC-client components.

3   When prompted for an installation directory, accept the default.

4   When prompted to restart the computer, leave the CD in the CD-ROM drive and select Yes. Restarting the PC updates environment variables and registry keys.

Configuring network connections for client products

Adaptive Server communicates with other Adaptive Servers, Open Server applications (such as Backup Server), and client software on your network. Clients can talk to one or more servers, and servers can communicate with other servers by remote procedure calls.

For Sybase products to interact with one another, each product needs to know where the others reside on the network. This information is stored in the *interfaces* file (*sql.ini* on Windows) or in the *libtcl.cfg* file (*libtcl64.cfg* on 64-bit versions).

Adding a server entry to the *libtcl.cfg* file

You use the *libtcl.cfg* files to specify an LDAP (Lightweight Directory Access Protocol) server name, port number, DIT base, user name, and password to connection to an LDAP server.

The default *libtcl.cfg* file is located in:

%SYBASE%\%SYBASE_OCS%\ini

In its simplest form, the *libtcl.cfg* file is in this format:

```
[DIRECTORY]
ldap=libdldap.dll ldapurl
```

where the *ldapurl* is defined as:

```
ldap://host:port/ditbase
```

When an LDAP server is specified in the *libtcl.cfg* file, the server information is accessible only from the LDAP server. Adaptive Server ignores the *interfaces* file. Open Client/Open Server applications that use the -I option at start-up override the *libtcl.cfg* file and use the *interfaces* file.

To use a directory service, you must:

1   Add the location of the LDAP libraries to the path environment variable for your platform.

2   Configure the *libtcl.cfg* file to use directory services.

   Use any standard ASCII text editor to:

   •   Remove the semicolon (;) comment markers from the beginning of the LDAP URL lines in the *libtcl.cfg* file under the *[DIRECTORY]* entry.

   •   Add the LDAP URL under the *[DIRECTORY]* entry. See *Configuring Adaptive Server Enterprise* for supported LDAP URL values.

Warning! The LDAP URL must be on a single line.

```
ldap=libdldap.dll ldap://host:port/ditbase??scope??
bindname=username password
```

For example:

```
[DIRECTORY]
ldap=libdldap.dll
ldap://huey:11389/dc=sybase,dc=com??one??
bindname=cn=Manager,dc=sybase,dc=com secret
```

3   Verify that the appropriate environment variable points to the required third-party libraries. The Netscape LDAP SDK libraries are located in *%SYBASE%\%SYBASE_OCS\lib3p*. The Windows PATH environment variable must point to this directory.

4   Once you have edited the *libtcl.cfg* file, use dsedit to add a server to the directory service:

    a   From the Windows task bar, select Start | Programs | Sybase | dsedit.

    b   Select LDAP from the list of servers, and click OK.

    c   Click Add New Server Entry.

    d   Enter:

       •   The server name – this is required.

       •   Security mechanism – optional. A list of security mechanism OIDs are located in *%SYBASE%\%SYBASE_OCS%\ini\objectid.dat*.

       •   HA server name – optional. This is the name of the high-availability failover server, if you have one.

    e   Click Add New Network Transport.

       •   Select the transport type from the drop-down list.

       •   Enter the host name.

       •   Enter the port number.

    f   Click OK two times to exit the dsedit utility.

For more information, see *Configuring Adaptive Server Enterprise* for your platform.

Adding a server to the *sql.ini* file

During installation, the srvbuild process adds entries to the *interfaces* file for your new Adaptive Server, Backup Server, Monitor Server, and XP Server.

To access a Sybase server through the *interfaces* file, each server must be included in the *interfaces* file on the client computer.

To add a server entry to the *sql.ini* file on your PC-client computer:

1   Using the Microsoft Windows task bar, select Start | Programs | Sybase | dsedit.

2   Click OK on the first screen to open the InterfacesDriver screen.

3   Select ServerObject from the menu, and click Add.

4   In the Input Server Name box, enter the name of the server for which you
     are creating an entry.

     Click OK.

5   In the Attributes column, double-click on the server address row you just
     added.

6   In the ProtocolNetwork Address, click Add.

7   From the drop-down list in the Protocol box, select TCP, NAMEPIPE, or
     SPX.

     By default, connections from client products to Adaptive Server are enabled
     through the Named Pipes and Winsock network protocols.

8   In the Network Address box, you can enter either the server name or the
     IP address, along with the server port number. For example:

```
machine_name, 4100
```

     or,

```
machine_name 4100
```

     where *machine_name* is the name of the computer, and *4100* is the port
     number the server is using to "listen" for clients.

Note The Windows platform now accepts TCP-style connection
information.

```
[SERVER]
MASTER=NLWNSCK,heuy,2222
QUERY=NLWNSCK,heuy,2222
```

or, the preferred format:

```
[SERVER]
MASTER=TCP heuy 2222
QUERY=TCP heuy 2222
```

or

```
[SERVER]
MASTER=TCP,heuy,2222
QUERY=TCP,heuy,2222
```

The preferred format is to use TCP and a space between the *host_name* and *port_number*, because it is supported across all platforms. You can edit the *sql.ini* with any standard ASCII text editor.

9   To find the IP address for a machine, use ypmatch. For example:

```
ypmatch 'hostname' hosts
```

where *hostname* is the name of the machine.

To find a server's listener server port number:

*   If the server is on another PC, check the *%SYBASE%\ini\sql.ini* file.

*   If the server is on a UNIX machine, check the *interfaces* file in *$SYBASE*.

If the host is on an HP machine and its server entry in the *interfaces* file looks like this:

```
SYBASE_125
    query tcp ether prod1 4100
    master tcp ether prod1 4100
```

To connect to this HP server, the complete entry in the sql.ini file reads:

```
SYBASE_125
query=TCP, prod1, 4100
```

Note To connect to a UNIX server, the entries you add to *sql.ini* must match the entries in the *interfaces* file on the UNIX system.

For instructions on using dsedit or dscp to modify *interfaces* file entries or to create new *interfaces* file entries for existing servers, see the *Utility Guide*.

For additional information on using dsedit on the client computer, see the *Open Client/Server Configuration Guide for Desktop Platforms* and the *Open Client/Server Supplement*.

For specific information about entries for connections to Monitor Server and Historical Server, see:

*   *Monitor Server User's Guide*

*   *Historical Server User's Guide*

Enabling TCP connections

To connect through TCP instead of Named Pipes and Winsock network protocols, you must enable the TCP connections:

1   From the Windows task bar, select Start | Programs | Sybase | OC OS Config.

2   Click the NetLibrary tab.

3   Select Windows NT or 98 as the platform and TCP as the protocol.

4   Click OK.

This process maps NLWNSCK to TCP in your Sybase network configuration file (*libtcl.cfg*).

You have completed installing Adaptive Server client plug-ins. To find out more information about setting up network communications, or to add optional functionality to Adaptive Server, see *Configuring Adaptive Server Enterprise*.

After you have installed the Sybase products on your system, see their accompanying documentation for configuration and administration issues.

Testing the Sybase Central installation

After you install Adaptive Server, Sybase Central, and the Java Runtime Environment, it is a good idea to test the installation and network connections:

1   Select Start | Programs | Sybase Central Java Edition from the Windows task bar to start Sybase Central.

2   Select Tools | Connect from the Sybase Central menu bar to activate a Sybase Adaptive Server login screen.

3   Log in using the default user ID, "sa", without a password. If you changed the password for Adaptive Server according to the post-installation instructions, you need to use the new password.

Figure 6-1: Sybase Adaptive Server Enterprise Logon screen
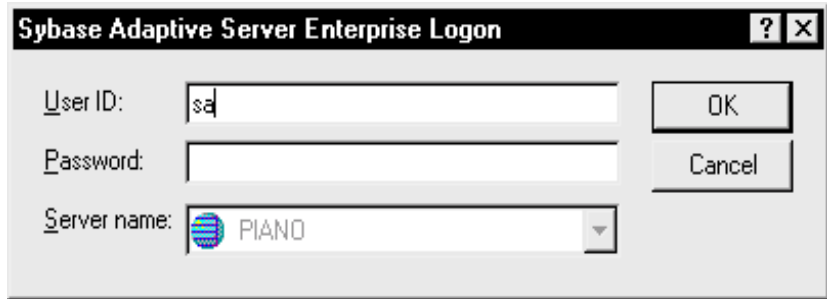
4   From the drop-down list, select the Adaptive Server to which you want to connect.

   If Sybase Central is installed on the same computer as the server, click the icon for the server.

   If prompted to start Adaptive Server, click OK. When the traffic light icon representing the server is green, the server is running.

Warning! Do not start XP Server from Sybase Central. Adaptive Server starts XP Server automatically when Adaptive Server receives the first request for an extended stored procedure.

---

5   To disconnect from a server, select Tools | Disconnect.

Note If a server is running on the same computer as Sybase Central, disconnecting does not stop it.

---

6   Exit Sybase Central.

If Adaptive Server fails any of these tests, see Chapter 10, "Troubleshooting." Before retrying the installation, follow the instructions in Chapter 5, "Post-Installation Tasks."

CHAPTER 7 **Starting and Stopping Servers**

This chapter describes how to start and stop Adaptive Server, Backup Server, Monitor Server, and XP Server.

Overview

After you create the Adaptive Server or upgrade Adaptive Server using srvbuild or sqlupgrade, Adaptive Server, Backup Server, and Monitor Server are running.

For information about running the Full-Text Search server, see the *Standard Full-Text Search Specialty Data Store User's Guide*.

The methods described here are used to start Adaptive Server, Backup Server, and Monitor Server after a shutdown for database maintenance, because of an operating system crash, or for some other reason. XP Server is started only when any XP command is issued through isql.

You can start Adaptive Server, Backup Server, and Monitor Server on UNIX either by:

- Entering commands from the UNIX command line, or

- Configuring your operating system so that the servers start automatically when the operating system starts

Starting servers

There are several ways to start and stop Sybase servers. This section describes how to start servers

Using the Runserver file

Each time a new Adaptive Server, Backup Server, or Monitor Server is installed, the srvbuild program creates a runserver file that contains the information required to restart that server. Runserver files are created in the *$SYBASE/$SYBASE_ASE/install* directory. For Adaptive Server, the runserver file is named *RUN_servername*, where *servername* is the name of the server.

For example, the runserver file name for an Adaptive Server named TEST is *RUN_TEST*. If the runserver file for your current Adaptive Server is named *RUNSERVER*, you must change the name to *RUN_TEST* during the upgrade process.

For Backup Server and Monitor Server, the runserver files are named *RUN_servername_back* and *RUN_servername_mon*, respectively, where *servername* is the name of the server.

Note Do not delete the runserver file that is created in *$SYBASE/$SYBASE_ASE/install*. This file is used to restart servers when you customize your installation. If you need the runserver file in another location, make a copy of the original runserver file in the new location.

Using the *startserver* command

To start a server from the command line enter:

```
$SYBASE/$SYBASE_ASE/install/startserver
[ -f runserverfile ]
```

where *$SYBASE/$SYBASE_ASE/install/startserver* is the full path name of the startserver utility, and *runserverfile* is the full path name of the runserver file (usually *$SYBASE/$SYBASE_ASE/install/RUN_servername*).

If your Adaptive Server is named SYBASE, the runserver file specification is optional.

You must have read and write permissions on the master device for an Adaptive Server to start it with startserver. See the *Utility Guide* for more information about startserver.

Although startserver runs Adaptive Server in the background, it does not return to the operating system prompt after restarting. After Adaptive Server has been restarted, press Return to return to the operating system prompt.

Using the *monserver* command

Use the monserver command to start Monitor Server from the UNIX command line. For details, see the *Monitor Server User's Guide*.

Stopping servers

Only the System Administrator has permission to issue a shutdown command. Using a shutdown command minimizes the amount of work that automatic recovery needs to do when the servers are restarted.

The preferred method of stopping Adaptive Server or Backup Server is to use the Transact-SQL shutdown command. For Monitor Server, use the sms_shutdown command.

Stopping Adaptive Server

To shut down Adaptive Server:

1   Use isql to log in to an Adaptive Server account with System Administrator privileges:

```
isql -Usa -Ppassword -Sserver_name
```

2   Enter the following command to shut down the server:

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

The default for the shutdown command is with wait. The with wait option allows Adaptive Server to finish executing SQL statements or procedures, perform a checkpoint in each database, disable new logins, and perform other shutdown tasks.

Issuing the shutdown command prints a message like this to the *stderr* file:

```
Server SHUTDOWN by request.The SQL Server is terminating
this process.
      CT-LIBRARY error:
    ct_results(): network packet layer: internal net
library error: Net-Library operation terminated due to
disconnect

      CT-LIBRARY error:

      ct_cancel(): network packet layer: internal net
library error: Net-Library operation terminated due to
disconnect
```

This is normal behavior. If the message indicates that Adaptive Server is waiting for processes to complete, and you need to stop Adaptive Server immediately, you can use the shutdown with nowait command. shutdown with nowait does not wait for currently executing statements to finish and does not perform checkpoints in every database.

Stopping Backup Server

To shut down a Backup Server:

1   Use isql to log in to a server with System Administrator privileges:

```
isql -Usa -Ppassword -Sserver_name
```

2   Enter the following command to shut down the specified Backup Server:

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

After you shut down a Backup Server, you must wait at least 30 seconds before restarting it.

Issuing the shutdown command prints a message similar to the following to the *stderr* file:

```
Backup Server: 3.48.1.1: The Backup Server will go down
immediately.
 Terminating sessions.
```

This is normal behavior. If a message indicates that Adaptive Server or Backup Server is waiting for processes to complete, and you need to stop Adaptive Server or Backup Server immediately, you can use the shutdown with nowait command. shutdown with nowait does not wait for currently executing statements to finish and does not perform checkpoints in every database.

Using shutdown with nowait for Backup Server can cause inconsistent or incomplete dumps and loads. Use this command only when necessary.

For more information on the shutdown command, see the *Reference Manual*.

To stop a Backup Server that is not configured as the logical Backup Server (SYB_BACKUP) for Adaptive Server:

• Change the logical Backup Server, SYB_BACKUP, to a different physical server:

```
1> sp_addserver SYB_BACKUP, NULL, common_name
2> go
```

where *common_name* is the name of the server in the *interfaces* (*sql.ini* on Windows) file.

• Perform the steps above to stop the server.

Stopping Monitor Server

You must shut down Monitor Server before restarting Adaptive Server. If Adaptive Server stops, make sure that Monitor Server is shut down before you restart Adaptive Server. Monitor Server must be shut down to release resources. Otherwise, Adaptive Server may not be able to allocate enough resources to restart.

To shut down Monitor Server:

• Log into the server:

```
isql -Smonitor_server -Usa -P
```

where *monitor_server* is the name of the Monitor Server with its extension.

• Issue the shut down command:

```
1> SMS_SHUTDOWN
2> go
```

If the Monitor Server heartbeat feature is in effect, Monitor Server automatically detects the termination of Adaptive Server within the specified period and shuts itself down. Therefore, before attempting to restart Adaptive Server after a shutdown, either wait for the automatic shutdown of Monitor Server to occur, or explicitly stop Monitor Server.

For more information on stopping Monitor Server, see the *Monitor Server User's Guide*.

Using the *kill* command

 Warning! Use the kill command in Adaptive Server and Backup Server only as a last resort.

When possible, use the Transact-SQL shutdown or shutdown with nowait command. Do not use kill with the -9 flag because it exits the server without running a checkpoint to ensure that all database changes are written to the database device. Adaptive Server may also exit without removing associated shared memory files and network handlers.

Because Adaptive Server and Backup Server are background processes, they can be killed from the operating system by their owner or by "root" with the UNIX kill command. The syntax is:

```
kill pid
```

where *pid* is the process identification of any dataserver or backupserver process, as determined by the showserver command. Killing one engine for a particular Adaptive Server kills all engines for that server.

If more than one Adaptive Server is running on the same system, you must be careful that the engine you kill is associated with the correct Adaptive Server. If your Adaptive Server is configured to use multiple engines (CPUs), each engine has an associated operating system process. The correct way to kill a multi-engine server is to specify the process ID for engine 0.

This showserver output shows the processes for a four-engine server:

```
showserver
```

UID	PID	PPID	C	STIME	TTY	TIME	COMD
jorge	3320	1	80	10:31:40	pts/4	302:15	dataserver -dteamster
jorge	3321	3320	80	10:31:45	pts/4	324:47	dataserver -ONLINE:1
jorge	3322	3320	80	10:31:45	pts/4	326:02	dataserver -ONLINE:2

| jorge | 3323 | 3320 | 80 | 10:31:45 | pts/4 | 328:56 | dataserver -ONLINE:3 |

This example shows four running dataserver processes with operating system process identifications (PID) 3320, 3321, 3322, and 3323. (dataserver is the executable form of the Adaptive Server program.)

Child engine processes for the dataserver have the -ONLINE: argument.

Each child engine has a parent process identification (PPID) that is equal to the process identification (PID) of the parent. In the example above, the PID of the parent server is 3320. The other three engines spawned by the parent process have the same PPID.

If the PPIDs appear to be unrelated, and there is more than one dataserver process, then more than one Adaptive Server is running on the system.

Shutdown and shared memory files

When Adaptive Server starts, it creates the following files in *$SYBASE* to store information about the shared memory segments that it uses:

SERVER_NAME.krg
SERVER_NAME.srg0

If Adaptive Server allocates more than one shared memory segment either during start-up or after start-up due to reconfiguration, it stores memory segment information in the file:

SERVER_NAME.srgNUM

where *NUM* is a number from 0 to 255. Note that Adaptive Server allocates a maximum of 257 shared memory segments.

Setting server start-up parameters

You can start Adaptive Server, Backup Server, or Monitor Server from the UNIX command line by using runserver files and startserver commands. You can move the Adaptive Server shared memory files by adding a flag to the runserver file.

Starting servers when the operating system restarts

This section describes how to configure your operating system for automatic restart of Adaptive Server and Backup Server when the system restarts.

On production systems, Adaptive Server, Backup Server, and Monitor Server should restart automatically when the UNIX operating system starts. This is accomplished by making an entry for the server in the operating system start-up script.

Monitor Server must be listed *after* Adaptive Server in the start-up script. For more information on starting Monitor Server during the operating system restart, see the *Monitor Server User's Guide*.

With HP-UX versions 10.0 and later, you cannot edit commands in the */etc/rc* file, so the System Administrator must create a script that executes when the operating system starts or shuts down.

Refer to the HP-UX UNIX manual page for "rc (1M)" for additional help. You can model your script after the HP-UX template file */sbin/init.d/template*.

After creating the start-up and shutdown script, place it in the directory */sbin/init.d*. Execution scripts placed in this directory have symbolic links to directories */sbin/rcn.d* where *n* is the system run level. The linked scripts in */sbin/rcn.d* are used to control the sequencing order of the execution scripts.

CHAPTER 8 **Upgrading Sybase Servers**

This chapter describes how to upgrade approved versions of Adaptive Server, Backup Server, and Monitor Server to version 12.5, and describes how to upgrade an Adaptive Server installation built on 32-bit Adaptive Server to run on a 64-bit Adaptive Server.

Note If you are migrating applications from one version of jConnect to another, see Chapter 5, "Migrating jConnect Applications," in the *jConnect for JDBC Programmer's Reference* for instructions.

Overview of the upgrade process

You can upgrade to Adaptive Server 12.5 from any of these versions:

- 11.5.x

- 11.9.x

For a server installation older than version 11.5.x, Sybase recommends that you upgrade to one of the versions above, then upgrade to version 12.5.

You can upgrade Adaptive Server from a 32-bit version to a 64-bit Version, but you cannot move from a 64-bit version to a 32-bit version. Likewise, you can only upgrade from an earlier version of Adaptive Server to a more recent version.

Only upgrades from 2K pages to 2K pages are supported. Changing the server schema from a 2K page to *n*K page size is a database migration, not an upgrade.

For information about migrating your database schema from a 2K page to a 4K, 8K, or 16K page, see "Migrating from 32-bit to 64-bit versions" on page 147.

Upgrading Adaptive Server consists of four processes:

• Installing the new server into it own installation directory.

> **Note** You must have both the old server and the new server to perform an upgrade.

• Performing the pre-upgrade checks on the old server using the preupgrade utility, from the new server installation.

• If necessary, fixing any problems that pre-upgrade process reports.

• Running the upgrade utility from the new server against the databases to update the underlying schema so that their structures are correct for the new server.

The preupgrade and upgrade utilities are internally called by the sqlupgrade utility.

Each new version of Adaptive Server contains different features that introduce new parameters, commands, reserved words, and so on. For this reason, the new Adaptive Server to which you are upgrading is responsible for preparing the old server for the upgrade.

The new server provides a utility, sqlupgrade, that runs various checks, such as reserved word checks, to determine how much space you must add to the old server to successfully upgrade the old server to the new.

As part of the pre-upgrade tasks, sqlupgrade scans all databases and catalogs and determines how much free space is required for each to upgrade successfully. Essentially, sqlupgrade searches for the largest catalog, then calculates the required free space by doubling the size of the largest catalog, and adding approximately 10 percent for logging the upgrade changes for each catalog.

During the pre-upgrade process, sqlupgrade returns informational messages as it checks the old server. You must fix all reported problems, and run sqlupgrade cleanly before beginning the upgrade process. Once the old server is eligible for upgrade, sqlupgrade shuts down the old server, starts the new server against the existing databases, and begins the upgrade process.

System catalog changes during upgrade

Adaptive Server 12.5 introduces support for wider columns, more columns, a larger number of user logins, and multiple logical page sizes. To support these extended limits, there have been several changes to existing system catalogs. See *What's New in Sybase Adaptive Server 12.5?* for a complete list of catalogs that are affected by the relaxed server limits.

Warning! As system catalogs are copied during the upgrade process, the size of these catalogs might cause the upgrade step to take a long time. Do not attempt to abort the process using Ctrl-C, as this can cause unknown corruptions in the catalogs.

Catalog changes that might affect existing applications

Any user-written stored procedures or applications that query system catalogs to obtain information must be changed to use the new datatypes for various columns. For instance, existing stored procedures and tools to regenerate table schema that look up information in syscolumns, will continue to work immediately following the upgrade. However, if new tables are created using the new limits, existing stored procedures are unable to retrieve the information from various catalogs.

For example, if a new table is created in with 300 columns, and an existing stored procedure declares a tinyint datatype to retrieve the column ID from syscolumns, this procedure cannot return the right information for this table because the table has more than 255 columns.

Similarly, other tools and procedures that access catalogs such as sysusers, syslogins, sysprotects, sysconstraints, and so on, must be updated to reflect the new column definitions. In general, all local variables in user applications and procedures need to be re-defined to match the datatype of the columns in system catalogs that are modified during the upgrade.

All Sybase stored procedures have been upgraded to reflect this change.

Pre-upgrade tasks

> **Note** Before you begin the upgrade process, install Adaptive Server 12.5 onto your system. See Chapter 4, "Installing Sybase Servers."

Before beginning any upgrade tasks, read the "Special Upgrade Instructions" section in the release bulletin for Adaptive Server.

To ensure a successful upgrade, review the following pre-upgrade tasks and perform them as necessary. Depending on the old server configuration, it may not be necessary to perform all pre-upgrade tasks.

All pre-upgrade tasks are performed on the old server. Pre-upgrade tasks include:

- "Check system and upgrade requirements" on page 106.
- "Test current applications and stored procedures" on page 108.
- "Reserved words" on page 108.
- "Verify that users are logged off" on page 110.
- "Check database integrity" on page 111.
- "Back up databases" on page 111.
- "Dump transaction logs" on page 112.
- "Prepare the database and devices for the upgrade" on page 112.
- "Turn off database options" on page 119.
- If you use replicated databases, read "Preparing to upgrade servers with replicated databases" on page 120.

Check system and upgrade requirements

> **Note** The sqlupgrade utility performs a pre-upgrade check of all database sizes and returns the value by which you need to increase database and devices sizes.

To verify that your system environment is configured correctly:

1   Verify that the computer on which you plan to upgrade the Sybase products meets the requirements described in Chapter 2, "Installation Requirements."

2   Determine whether your server supports an upgrade to Adaptive Server 12.5.

    To determine the version level of your current Adaptive Server, enter the following command after login in to your server using isql:

    ```
    1> select @@version
    2> go
    ```

    You can upgrade to Adaptive Server 12.5 from any of these versions:

    •   11.5.x

    •   11.9.x

    If you are running a version of Adaptive Server that is earlier than those listed here, you must upgrade to one of these versions before upgrading to 12.5.

3   Set the environment variables to the values described in Chapter 4, "Installing Sybase Servers."

4   Verify that you unloaded Adaptive Server 12.5 in a *different* directory than your previous Adaptive Server installation.

    If you overwrote your previous server installation:

    a   Restore it from your most recent backup.

    b   Reinstall the product files for Adaptive Server in a different directory.

    c   Continue with the upgrade.

5   Verify that your operating system is at the proper version level and has all operating system patches needed for Adaptive Server. See the *Release Bulletin* for your platform.

Check RUN_*server* file location

Verify the name and location of the *runserver* file. Be sure the *runserver* file for your current server is located in:

•   *$SYBASE/install* directory for versions 11.5.x and 11.9.x

•   *$SYBASE/$SYBASE_ASE/install* for version 12.0.x

107

Verify that the file is still named *RUN_servername*, where *servername* is the name of the old server. The *servername* must appear exactly as it appears in the *interfaces* file.The *RUN_servername* file for a server SYBASE is called *RUN_SYBASE*. If the *RUN_servername* file for your current Adaptive Server is named, you must change the name during the upgrade process.

Test current applications and stored procedures

If you have critical applications and stored procedures, perform an upgrade in a test environment, so that you can determine any adjustments you may need to make in your production environment before upgrading it.

Determine which Sybase products should be running. If you are upgrading Adaptive Server, the previously installed version of the server *must* be running. If you are upgrading Backup Server, Historical Server, Monitor Server, or XP Server, those servers must *not* be running.

Procedure text is required for upgrade

If you created any compiled objects or stored procedures in a pre-11.9.3 server, then removed the syscomments text from sybsystemprocs, the upgrade will succeed but you will encounter runtime problems. To upgrade system procedures, the system procedure text must be available in syscomments. See "Upgrading compiled objects with dbcc upgrade_object" on page 139.

Reserved words

Reserved words are pieces of SQL syntax that have special meaning when used as part of a command.

Transact-SQL does not allow words that are part of command syntax to be used as identifiers, unless they are enclosed in quotation marks, as explained in "Using quoted identifiers" on page 110. If you are upgrading Adaptive Server, and the identifiers in your user databases match new reserved words, errors can result when you run queries, stored procedures, or applications that use these identifiers.

Warning! If a user database name is a reserved word, upgrade fails when it tries to upgrade the database. You must use sp_renamedb to change the name of any user database that is a reserved word before performing the upgrade.

Conflicts between object names do not prevent the upgrade process from completing. However, applications that refer to conflicting object names may not work after the upgrade. Rename all objects that use reserved words.

As part of the pre-upgrade process, sqlupgrade can perform the reserved word check for you. See the *Reference Manual* for a complete list of reserved words.

Note If you change an object name, change applications and stored procedures that refer to that object.

Running a reserved word check

If you plan to use sqlupgrade to perform the upgrade process, you do not have to check for reserved words as a separate step. In sqlupgrade, you can check for reserved words at the start of the upgrade process, and then continue with the upgrade if no problems are found.

If you plan to use sqlupgraderes to perform the upgrade process using a resource file, you can run sqlupgrade first to identify reserved word conflicts and other potential upgrade eligibility problems. You can specify that you do not want the upgrade to proceed even if no problems are found.

If you use either sqlupgrade or sqlupgraderes, they automatically install new reserved words and sp_checkreswords, a stored procedure used to detect and display any identifiers in your existing databases that conflict with reserved words in the new database. You can then run the new version of sp_checkreswords at anytime while preforming pre-upgrade tasks.

Note You must use the sp_checkreswords stored procedure from the new version of Adaptive Server to verify that the old installation of Adaptive Server does not use any reserved words that were introduced with the new server.

Checking for reserved words generates a list of identifiers that conflict with reserved words, and the owners of those identifiers, in the file *$SYBASE/$SYBASE_ASE/init/logs/sqlupgradeMMDD.nnn*. Review this file to determine which identifiers must be changed.

Addressing reserved words conflicts

If any database names are reserved words, you must use sp_renamedb to change the database names before you can upgrade. Use sp_dboption to set the database to single-user mode, and then run sp_renamedb, specifying the new name. See the *Reference Manual* for more information on these procedures.

If other identifiers are reserved words, you can:

*   Use sp_rename to change the name of the object, before or after the upgrade.

*   Use quoted identifiers. See "Using quoted identifiers" on page 110 for more information.

Run sp_checkreswords in master and in each user database to display the names and locations of conflicting identifiers.

For more information about sp_rename and sp_checkreswords and methods for avoiding reserved word conflicts, see the *Reference Manual*.

Using quoted identifiers

You can enclose the identifiers that are reserved words in double quotation marks and invoke the quoted_identifier option of the set command in procedures and queries that include the reserved words. The set quoted_identifier option tells Adaptive Server to treat any character string enclosed in double quotation marks as an identifier.

To avoid reserved word conflicts, all users on the server must invoke the quoted_identifier option in all stored procedures and queries that include the reserved words.

Verify that users are logged off

To verify that users are logged off of the system:

1   Log in to your current server as the Sybase System Administrator:

```
isql -Usa -Ppassword -Sserver_name
```

2   Enter:

```
1> sp_who
2> go
```

This command returns a list of all users and processes currently on the server.

3   Log off all users on the server.

The upgrade will not work if you start the server in single-user mode.

Check database integrity

Check logical and physical database consistency by running the following commands in each database. Use isql to log in to the server, and run:

*   dbcc checkalloc
    ```
    dbcc checkalloc [(database_name [, fix | nofix] )]
    ```

*   dbcc checkdb
    ```
    dbcc checkdb [{database_name [, skip_ncindex]) ]
    ```

*   dbcc checkcatalog
    ```
    dbcc checkcatalog [(database_name)]
    ```

For a description of the dbcc functions, see the *System Administration Guide*.

If you encounter any errors while running these commands, resolve them before continuing the upgrade. For help with resolving errors, see the *Error Messages and Troubleshooting Guide*.

If you are upgrading from Adaptive Server 11.5.x or later, and you have set up your system to run dbcc checkstorage, you can use that command instead of the other dbcc commands to check database consistency. For details on using dbcc checkstorage, see the *System Administration Guide*.

If a database is marked "suspect," drop that database with the dbcc dbrepair (*database_name*, dropdb) command.

Back up databases

Use the dump database command to back up your databases. If you encounter any problems during the upgrade, you may need these dumps to recover the previous installation. Be sure to dump the master database.

> **Warning!** *The upgrade process permanently modifies your existing Sybase databases.* Back up all your databases, including master, sybsystemprocs, and model before beginning the upgrade so you can restore them if necessary.

For information on backing up and recovering databases, see the *System Administration Guide*.

Dump transaction logs

The upgrade process may consume substantial space in the transaction log. Before beginning the upgrade process, run the dump transaction command in each database to free up log space. See Chapter 27, "Backing Up and Restoring User Databases" in the *System Administration Guide*.

Prepare the database and devices for the upgrade

Make sure that master is the default database for the "sa" user. The upgrade cannot complete successfully with a different database as your default. For information about specifying the default database, see the *System Administration Guide*.

If you ran the pre-upgrade option in sqlupgrade, the utility scanned the system catalogs and calculated how much additional space you needed to perform the upgrade. If you did not run the pre-upgrade check, you must manually calculate how much free space you must add to your system catalogs and databases.

As a general rule, you must double the size of the largest catalog that you are going to upgrade, and add approximately 10 percent more to calculate the free space required for the upgrade to succeed.

For example, if you have a large number of stored procedures or compiled object in any database, the syscomments and sysprocedures catalogs will require additional space.

> **Note** Sybase recommends that you use preupgrade to determine how much free space is required for the upgrade.

An installation of Adaptive Server 12.0, without any additional user-defined data creates the following:

- tempdb with 2MB

- model with 2MB

- master database with 6MB

If you are upgrading an Adaptive Server installation that meets these minimum parameters, you must increase:

- tempdb to 4MB

- model to 4MB

- master database to 8MB

Use sp_help to determine the size of your existing databases. If any database does not have the space requirements as reported by preupgrade, use alter database to increase the size.

❖ **Increasing default database sizes**

To determine the size of each databases in the old installation, log in to Adaptive Server using isql:

```
isql -Usa -Ppassword -Sserver_name
```

Enter:

```
1> sp_helpdb database_name
2> go
```

where *database_name* is the name of the system database you are checking.

To enlarge the master, tempdb, and model databases:

1   In isql, use alter database to increase the size of the master database. For example:

```
1> use master
2> go
1> alter database master on master=x
2> go
```

This example increases the size of the master database; *x* is the number of megabytes of space added to the existing database size.

2   Repeat this step to increase the size of each of the tempdb and the model databases. For example:

```
1> alter database tempdb on master=x
2> go
1> alter database model on master=x
```

113

```
2> go
```

> **Note** You must increase the size of tempdb before increasing the size of model.

3   Verify the size of each database. Enter:

```
1> sp_helpdb database_name
2> go
```

where *database_name* is the name of the system database you are checking.

Create a *sybsystemdb* database

In versions 11.5.x and 11.9.x, the sybsystemdb database was required only for servers using two-phase commit transactions. Beginning with version 12.0, all servers must have a sybsystemdb database. Adaptive Server uses this database for tracking transactions and during recovery. In addition, it is used for applications using two-phase commit and distributed transaction management (DTM).

sybsystemdb does not need to be located on the master device.

If you have a *sybsystemdb* database

If you have a sybsystemdb and the database is at least 4MB, you do not need to make any changes. If the database is smaller than 4MB, increase the size with alter database.

If you do not have a *sybsystemdb* database

If you do not have a sybsystemdb, and will not be using two-phase commit or DTM applications, create a sybsystemdb with a minimum of 4MB.

If you will be using two-phase commit or DTM, you can either:

*   Create a minimum-sized sybsystemdb for upgrade and expand it later, or

*   Create a sybsystemdb of 5 to 20MB, depending on your expected usage. Approximately 25 percent of the database should be data storage, and 75 percent should be log storage.

The following example initializes data and log devices, and creates a 20MB sybsystemdb:

```
1> disk init name = "sybsystemdb_data",
```

```
2> physname = "/sybase/data/sybsystemdb_data",
3> vdevno = 8,
4> size = 2560
5> go
1> disk init name = "sybsystemdb_log",
2> physname = "/sybase/data/sybsystemdb_log",
3> vdevno = 9,
4> size = 7680
5> go
1> create database sybsystemdb on sybsystemdb_data=5
2> log on sybsystemdb_log = 15
3> go
```

sybsystemprocs

Note sysprocsdev is the default system name for this device. However, it is frequently referred to as the sybsystemprocs device, as in the Adaptive Server attribute screen, since it stores the sybsystemprocs database.

Verify that the sybsystemprocs database is large enough. For an upgrade, the recommended minimum size for sybsystemprocs is the larger of 105MB, or enough free space to accommodate the existing sybsystemprocs database, and the largest catalog that is to be upgraded, plus an additional 10 percent of the largest catalog's size. The additional 10 percent is for logging upgrade changes.

You may need more space if you are adding user-defined stored procedures. 105MBs accommodate additional internal data structures, but does not account for the possibility of a large number of user-defined system procedures.

If your sybsystemprocs database does not meet these requirement and you have enough room on the device to expand the database to the required size, use the alter database command to increase the database size.

Use sp_helpdb to determine the size of the sybsystemprocs database:

```
1> sp_helpdb sybsystemprocs
2> go
```

Use sp_helpdevice to determine the size of the sysprocsdev device:

```
1> sp_helpdevice sysprocdev
2> go
```

If the db_size setting is less than the required minimum, you must increase the size of sysprocdev.

115

Increasing the size of the *sybsystemprocs* database

If your current sybsystemprocs database does not have the minimum space required, you have two options for creating a new database with sufficient space for the upgrade:

• You can drop the old database and the device and create a new sysprocsdev device, or

• You can leave the database and old device alone and add a new device large enough to hold the additional megabytes, and alter the sybsystemprocs onto the new device. Sybase recommends expanding the current device.

To enlarge the sybsystemprocs database:

1 If you do not have a current backup, create one.

2 In isql, use alter database to increase the size of the sybsystemprocs database. For example:

```
1> use master
2> go
1> alter database sybsystemprocs on sysprocsdev=40
2> go
```

In this example, "sysprocsdev" is the logical name of the existing system procedures device and "40" is the number of megabytes of space to add.

If the system procedures device is too small, you may receive a message similar to the following when you try to increase the size of the sybsystemprocs database:

```
Could not find enough space on disks to extend
database sybsystemprocs
```

If there is space available on another device, you can expand sybsystemprocs to a second device, or initialize another device that is large enough. For instructions on creating a larger sybsystemprocs device, see "Increasing device and database capacity for system procedures" on page 117.

3 To verify that Adaptive Server has allocated more space to sybsystemprocs, issue:

```
1> sp_helpdb sybsystemprocs
2> go
```

When the system procedures database is large enough to accommodate the increased size of the sybsystemprocs database, continue with the other pre-upgrade tasks.

Increasing device and database capacity for system procedures

If you cannot fit the enlarged sybsystemprocs database on the system procedures device, increase the size of the device and create a new database.

This procedure involves dropping the database. For more information on drop database, see the *Reference Manual*.

 Warning! This procedure removes all stored procedures you have created at your site. Before you begin, save your local stored procedures using the defncopy utility. See the *Utility Guide* for more information on Adaptive Server utilities.

To create a larger system procedures device (sysprocsdev):

1   Determine which device or devices you need to remove.

 Warning! You must *not* remove any device that is in use by database other than sybsystemprocs, or you will destroy that database.

```
select d.name, d.phyname
from sysdevices d, sysusages u
where u.vstart between d.low and d.high
and u.dbid = db_id("sybsystemprocs")
and d.status & 2 = 2
and not exists (select vstart
    from sysusages u2
    where u2.dbid != u.dbid
    and u2.vstart between d.low and d.high)
```

where *d.name* is the list of devices to remove from sysdevices and *d.phyname* is the list of files to remove from your computer.

The "not exists" clause in this query excludes any devices that are used both by sybsystemprocs and other databases.

Note the names of the devices; you will need them in the coming steps.

2   Drop sybsystemprocs:

```
use master
go
```

117

```
drop database sybsystemprocs
go
```

3   Remove the device or devices using sp_configure:

```
sp_configure "allow updates", 1
go

delete sysdevices
where name in ("d.name")
go

sp_configure "allow updates", 0
go
```

where *d.name* is the name of each devices returned in step 1.

> **Note** Each device name must have quotes. For example, "*devname1*",
> "*devname2*", and so on.

If any of the named devices are OS files rather than raw partitions, use the
appropriate OS commands to remove those files.

4   Remove all files for the list of *d.phyname* returned in step 1.

> **Note** File names may not be complete path names. If they are relative
> paths, they will be relative to the directory from which your server was
> started.

5   Find another existing device that meets the requirements for additional
     free space, or use a disk init command similar to the following to create an
     additional device for sybsystemprocs:

```
1> use master
2> go

1> disk init
2> name = "sysprocsdev",
3> physname = "$SYBASE/work/sysproc.dat",
4> vdevno = 9,
5> size = 51200
6> go
```

where *$SYBASE/work/* is the full, absolute path to your system procedures device. You cannot use the environment variable *$SYBASE* in this command.

Note The number for vdevno must be available. For information about determining whether vdevno is available, see the *System Administration Guide*.

The size you provide should be the number of megabytes of space needed for the device multiplied by 512. disk init requires the size to be specified in 2K pages. In this example, the size is 105MB (105 x 512 = 53760). For more information on disk init, see the *Reference Manual*.

6   Create a sybsystemprocs database of the appropriate size on that device, for example:

```
1> create database sybsystemprocs on sysprocsdev =
105
2> go
```

7   Run the installmaster script in the *old* Sybase installation directory, as follows:

```
isql -Usa -Ppassword -Sserver_name -i/old_dir/
scripts/installmaster -oinstallmaster.out
```

You must reinstall the system stored procedures for use later in this procedure.

Turn off database options

To turn off database options:

1   Run sp_helpdb to show the options that are set for each database.

2   Use sp_dboption and the results from sp_helpdb to turn off all options *except* select into/bulk copy on tempdb.

   For example, to turn off the trunc log on chkpt database option, enter:

```
sp_dboption sybsystemprocs, 'trunc log on chkpt',
false
```

   Turn off the trunc log on chkpt database option to avoid upgrade failure.

For more information about using these system procedures, see the *Reference Manual*.

Note The database options for master cannot be changed and are not disabled.

Disable auditing

To upgrade a server that contains a database in which auditing is activated, make sure that auditing is disabled before starting the upgrade:

1   Turn off auditing before upgrading.

    a   Verify that all Adaptive Server users are logged off.

    b   Disable auditing with the following Transact-SQL command:

```
sp_auditoption 'enable auditing', 'off'
```

2   Record system procedure audit options before upgrading.

When upgrading, all previous versions of system procedures are overwritten when sybsystemprocs is populated. Record the audit options for all system procedures. After the upgrade is complete, shut down and restart the server. Reenable auditing using:

```
sp_configure auditing, 1
```

You can reenter the audit options, using sp_audit. For more information, see the *System Administration Guide*.

Disable disk mirroring

Use the disk unmirror command to disable disk mirroring for your current Adaptive Server. For more information about this command, see the *Reference Manual*.

Preparing to upgrade servers with replicated databases

After upgrade, you will no longer be able to scan any part of the transaction log that existed before the upgrade, so you must follow the following process if your server contains replicated primary databases (this includes replicated RSSDs). The following procedure will help to ensure that all replicated data from a replicated database has made it safely to the replicate database.

Warning! It is not sufficient to just get the replicated data into the Replication inbound queue, because the inbound queue cannot be rebuilt after the upgrade.

The procedures described here do not upgrade Replication Server itself. For information on upgrading Replication Server, see your Replication Server documentation.

The database upgrade procedure consists of the following activities:

• Suspending transaction processing and replication activities.

• Draining transaction logs for primary databases.

• Draining the Replication Server System Database (RSSD) log.

• Disabling the log truncation point.

After upgrading to version 12.5, complete the post-upgrade tasks to reenable database replications functions.

For more information, see the *Replication Server Reference Manual* and the *Replication Server System Administration Guide*.

Warning! As a safeguard, perform a dump database and a dump transaction before executing the procedures in the following sections.

To determine whether your existing server contains replicated databases:

1   Connect to the Server you are upgrading via isql.

2   Run the following command in each database (including system databases):

```
1> dbcc gettrunc
2> go
```

3   If the command returns "1" for "ltm_trunc_state" in any database, replication is enabled in that database.

Suspending transaction processing and replication activities

To suspend replication of and transaction activity in the databases:

1   Verify that subscriptions that are being created by the create subscription command, with primary data in the databases being upgraded, have reached the "valid" state at the primary Replication Server before you begin the upgrade. Use the check subscription command to find the "valid" state.

    Defer upgrade while the subscriptions are being created so that Replication Server does not interfere with the upgrade process by accessing the database being upgraded.

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

2   Execute rs_helproute in each Replication Server System Database (RSSD) being upgraded.

    The status of all existing routes should be "Active". If any route is not active, resolve the route before continuing. See the Replication Server documentation for help in diagnosing and fixing the problem with the route, and then go to step 3.

3   Shut down all applications that use the databases you are upgrading.

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

5   Suspend all DSI connections to the non-RSSD databases you are upgrading by entering the following command in Replication Server for each database:

```
1> suspend connection to dataserver.database
2> go
```

6   Leave the DSI connections to the RSSD databases running.

Draining the transaction logs for primary databases

For each primary database you are upgrading, you need to ensure that Replication Server completely processes the pre-upgrade log.

To drain the transaction logs:

1   Wait for all remaining transactions to be replicated.

2   Run the following Replication Server command:

```
1> admin who, sqm
2> go
```

3   Find the entry that corresponds to the inbound queue for this database by looking for the Info field for the queue_number:queue_type entry. For an inbound queue, the queue type is 1. Note the Last Seg.Block entry for the queue.

4   Open a queue dump file by executing the following Replication Server command:

```
1> sysadmin dump_file, "file_name"
2> go
```

where *file_name* is the name of the file to which you will dump the queue.

5   Use isql to update one row in a single replicated table in the primary database:

```
1> update table set column = column
2> where key = unique_value
3> go
```

The update command helps to track whether all modifications to the replicated database have been sent to the Replication Server.

Note In Replication Server 10.1 or later, choose a table that does not use the replicate minimal columns clause, or use the alter replication definition command...replicate all columns command to change the replication definition before updating the row. If you alter the replication definition, be sure to change it back after you complete this upgrade procedure.

---

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

7   Execute the following Replication Server command to dump the last block of the inbound queue to the dump file you created in step 3:

```
1> sysadmin dump_queue, queue_number,
2> queue_type, last_seg, block, 1
3> go
```

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

8   Examine the dump file to make sure it contains the transaction that corresponds to the update you performed in step 4. (You can use Notepad to examine the file.)

9   Repeat steps 5–7 until the transaction that corresponds to the update is in the dump file.

10   Log into the Replication Server and suspend the Log Transfer connection from that database:

```
1> suspend log transfer from server.database
2> go
```

11   If you are using Rep Agent, log into the ASE server, and stop the Rep Agent:

```
1> use database
2> go

1> sp_stop_rep_agent database
2> go
```

12   If you are using LTM, shut down the LTM.

After draining the transaction logs, do not allow any other activity in the databases. If activity does occur, you will need to redrain the logs.

Draining the RSSD Transaction Log

If the Replication Server has routes to other Replication Servers, you must ensure that Replication Server processes all transactions in the RSSD transaction log before you upgrade the databases.

To see whether the transaction log has been processed completely, create a replication definition in the primary Replication Server and then watch for it to appear in the replicate Replication Server's RSSD. When the replication definition is in the replicate RSSD, you can assume that the log is processed fully.

To ensure that the RSSD log is processed:

1   Log into the primary Replication Server and create a temporary replication definition:

```
1> create replication definition rep_def_name
2> with primary at dataserver.database
3> (column_a int)
4> primary key (column_a)
5> go
```

The data server and database names must be valid, but the replication definition does not have to reference an actual table.

2   Log into the replicate RSSD (not the primary RSSD) and execute the following query to find out if the replication definition has arrived from the primary RSSD:

```
1> select * from rs_objects
2> where objname = "rep_def_name"
3> go
```

If this select statement returns rows, the last replication definition created in step 1 has been sent successfully to the replicate RSSD. This means that the transaction log has been drained.

3   Log into the replicate Replication Server and suspend the Log Transfer connection from the primary RSSD:

```
1> suspend log transfer from server.database
2> go
```

4   If you are using Rep Agent, log into the ASE server, and stop the Rep Agent:

```
1> use database
2> go

1> sp_stop_rep_agent database
2> go
```

5   If you are using LTM, shutdown the LTM.

Disabling the secondary truncation point

When you upgrade a primary database, the Rep Agent or the Log Transfer Manager (LTM) must not be running, and the Secondary truncation point should be turned off for the duration of the upgrade. The Rep Agent or Log Transfer Manager should already be shutdown (from the previous steps).

For each primary database and replicated RSSD, disable the secondary truncation point:

1   If this is a replicated RSSD, log into the Replication Server of the RSSD, and issue:

```
1> sysadmin hibernate_on, 'Replication Server'
2> go
```

2   Disable the secondary truncation point in the Adaptive Server database that is being upgraded by issuing:

```
1> use database
```

125

```
2> go
1> dbcc settrunc('ltm', 'ignore')
2> go
```

If the dbcc settrunc command fails, make sure that the Rep Agent or LTM are not running. When the Rep Agent and LTM are disabled, repeat this step.

Upgrading to Adaptive Server 12.5

To assist with the upgrade process, Sybase recommends that you set the OLDSYBASE environment variable to point to the old installation of Adaptive Server. If you set the OLDSYBASE environment variable to the old installation of the Adaptive Server, the upgrade utility will automatically populate input fields with the correct information as you perform the upgrade.

Back up all existing Sybase installations. Installing Adaptive Server files into an existing Sybase installation directory overwrites the previous installation.

Warning! If you install Adaptive Server files into the same directory as the currently installed version, the new version overwrites the old one. If this occurs, the upgrade cannot complete, because you must have both the old and the new versions of Adaptive Server to perform an upgrade.

Use either of the following methods to upgrade Adaptive Server:

*   sqlupgrade – provides an X-Windows or Motif GUI for the upgrade process. See "Using sqlupgrade" on page 127.

*   sqlupgraderes – provides a noninteractive, file-based interface for the upgrade process. sqlupgraderes allows you to upgrade Adaptive Server using a resource file. This is useful for customer sites that want to upgrade many similar servers. See "Using sqlupgraderes" on page 129.

Note Backup Server, Monitor Server, and XP Server cannot be upgraded using sqlupgraderes.

Using *sqlupgrade*

sqlupgrade verifies that the server is eligible for the upgrade. Perform all pre-upgrade tasks before running sqlupgrade.

To upgrade using sqlupgrade:

1   Make sure your current server is running.

2   Verify that your SYBASE environment variable points to the location of the new Adaptive Server software files you just unloaded.

You also can set the OLDSYBASE environment variable to the location of the server you are upgrading, to avoid typing this path when you run sqlupgrade.

3   Verify that you have performed the tasks mentioned on the screen, before beginning the upgrade.

> **Note** sqlupgrade automatically copies the *interfaces* file entry of the server you are upgrading from the old *interfaces* file to the new *interfaces* file.

4   Execute sqlupgrade in either of the following ways:

   •   At the UNIX prompt, enter:

        $SYBASE/$SYBASE_ASE/bin/sqlupgrade

   A sqlupgrade screen is displayed.

   •   Alternatively, you may access sqlupgrade through the asecfg utility. Then click on the Upgrade an Existing Server button.

5   Click OK.

6   The Specify Sybase Directories screen of sqlupgrade is displayed. Provide the installation directory location for both the new server version and your current server.

If the SYBASE environment variable is set, sqlupgrade displays the value of the SYBASE environment variable as the directory location for the new server version.

If the OLDSYBASE environment variable is set, sqlupgrade displays that value as the directory location for your current server.

7   Enter the name of the component directory that contains the new server version. For example, *ASE-12_5*.

8   Enter the full path of the old Sybase release directory. For example, */opt/sybase-12.0*.

If the OLDSYBASE environment variable is set, sqlupgrade displays that value as the directory location for the old server.

9   In the field, "Enter the name of the component directory which contains the old version" enter the Adaptive Server component directory. For example, *ASE-12_0*.

If you are upgrading from a version earlier than 12.0, leave this field blank. Beginning with Adaptive Server 12.0, the directory structure changed.

10 Click OK. The Adaptive Server Selection screen displays.

11 From the list of server names provided, select the server you want to upgrade.

If the server you are upgrading is not running, sqlupgrade prompts you to start the server. The server you are upgrading *must* be running.

12 Click OK. The SA Password screen is displayed.

13 Enter the password for the System Administrator (sa) login.

14 Click OK. The Specifying Upgrade Options screen is displayed.

15 Specify an upgrade option.

The upgrade eligibility tests verify that your current server meets the requirements for upgrading to the new version. (The online help provides a partial list of what the upgrade eligibility tests check.) If your current server fails the upgrade eligibility test, you must fix the problem.

Note Adaptive Server 11.9.x did not automatically create a sybsystemproc database and device. Unless you created one, the eligibility test will fail. The Back button launches the "Specify sybsystemprocs" screen.

Enter the full path to the sybsystemproc device and 105MB for the sybsystemproc database. Click OK to continue with the eligibility test.

---

16 Click OK to upgrade your server.

sqlupgrade displays the Status Output screen. View the Status Output screen for the completion status and informational messages about the upgrade process.

Warning! Do not interrupt the upgrade, and do not try to connect to Adaptive Server or run any stored procedures while the upgrade is in progress.

You can also check the progress of the upgrade by viewing the sqlupgrade log in *$SYBASE/$SYBASE_ASE/init/logs/sqlupgradeMMDD.NNN*.

After all pre-upgrade checks are successful, the old server is shutdown and the new Adaptive Server dataserver binary is started on the old master device. sqlupgrade internally runs the upgrade binary to perform the upgrade.

If the upgrade is successful, a "Done" message is displayed in the Status Output screen.

Note On some UNIX platforms, the sqlupgrade utility may not return you to the command prompt even though the upgrade has completed successfully. To exit the utility, force an exit by pressing Ctrl + C.

---

17 The upgrade process:

•   Creates the *RUN_servername* file

•   Runs *installmaster* and *installjconnect* scripts

When the upgrade completes successfully, you can:

•   Click OK to specify another server to upgrade, or

•   Exit sqlupgrade, and go to "Post-upgrade tasks" on page 131.

•   See "Upgrading compiled objects with dbcc upgrade_object" on page 139.

Using *sqlupgraderes*

You can upgrade Adaptive Server in noninteractive mode by using values from a resource file that defines the attributes for the server to be upgraded.

To upgrade an Adaptive Server using a resource file, either:

1   Edit the sample resource file included in your Adaptive Server distribution in *$SYBASE/$SYBASE_ASE/init/sample_resource_files/upgrade.rs*. Use an ASCII text editor to edit the resource file as described in the Appendix, "Alternative Installation Methods."

129

2   Run the sqlupgrade utility. This utility creates a new file containing the values you specified for that upgrade session, and writes it to:

$SYBASE/$SYBASE_ASE/init/logs/sqlupgradeMMDD.NNN-server_name.rs

where *server_name* is the name of the upgraded server.

> **Warning!** Be sure to change device names in the resource file, if necessary, before executing the resource file using sqlupgraderes.

For details on the attributes used in a resource file, see "Resource file attributes for upgrading Adaptive Server" on page 130.

Resource file attributes for upgrading Adaptive Server

Table 8-1 shows the resource file attribute names, their default values, and other options, which you use in the resource file for upgrading Adaptive Server. The sample resource file in the *$SYBASE_ASE/init/sample/upgrade* file included in your Adaptive Server distribution shows examples of these attributes.

All attributes are *required*. And all values are case sensitive.

If the resource file you are modifying was created by sqlupgrade, the prefix of the attribute name may vary. sqlupgraderes ignores this prefix when processing the resource file.

Table 8-1: Resource file attributes for upgrading Adaptive Server

Attribute	Default value [other options]
sybinit.release_directory	$SYBASE [*path_name_of_old_srv*]
sybinit.product	sqlsrv
sqlsrv.server_name	*server_name*
sqlsrv.new_config	no (required value)
sqlsrv.sa_login	*current_login*
sqlserv.sa_password	*current_password*
sqlsrv.do_upgrade	yes (required value)

Upgrading using *sqlupgraderes*

When you have finished editing your resource file, you can run the file using sqlupgraderes.

Note sqlupgraderes is a command-line tool; therefore, unlike sqlupgrade, it
does not require any X libraries.

To execute sqlupgraderes, at the UNIX prompt, enter:

```
$SYBASE/$SYBASE_ASE/bin/sqlupgraderes -r resource_file
```

where *resource_file* specifies the resource file containing the attributes that
describe the server to upgrade.

Post-upgrade tasks

After you have upgraded to the new version of Adaptive Server, perform the
following tasks to make sure your new Adaptive Server is up and running.

Note After upgrade, there is no need to run update statistics on any tables.
Upgrade process does not cause any changes in existing statistics.

Verify that servers are running

1   To find out whether Adaptive Server is currently running, try to log in to
    the server, using isql:

```
isql -Usa -P -Sserver_name
```

If the server is running, you see the isql prompt:

```
1>
```

You can also use the showserver command to determine whether any
Adaptive Server processes are running. For example:

```
$SYBASE/$SYBASE_ASE/install/showserver
```

2   To verify that you are at the new version level, connect to Adaptive Server
    via isql and run the following commands:

```
1> select @@version
2> go
```

Look for "12.5" in the version string.

```
1> sp_configure "upgrade version"
2> go
```

sp_configure should return the Run Value "12500".

Restore functionality in Adaptive Server

To reset or enable options or functionality you had to change before the upgrade:

1   Reset the configuration parameters.

    If you changed any configuration parameters before upgrade, use sp_configure to set them back to their previous values.

2   Use sp_dboption to reset any database options you disabled before the upgrade.

3   After you complete the upgrade, but before you use the upgraded Adaptive Server, verify that all scripts developed at your site point to Adaptive Server 12.5.

4   Procedure cache allocation after upgrade.

    This section is for information only. Adaptive Server ensures that procedure cache size after upgrade is the same as before upgrade, unless the before upgrade size was less than the default value.

    In Adaptive Server 12.5, procedure cache percent has been changed to procedure cache size. Adaptive Server now allocates memory dynamically and determines the size of the procedure and data caches in terms of megabytes. During the upgrade process, Adaptive Server converts the procedure cache percent to an absolute value and sets procedure cache size based on this value.

    For example, if procedure cache percent is configured as 20 percent of 200MB of memory, then the absolute value of procedure cache percent is 40MB. During the upgrade process, Adaptive Server converts procedure cache percent at 20 percent to procedure cache size at 40MB. If the value for procedure cache size is less than the default value, Adaptive Server sets procedure cache size to the default value. For example, if procedure cache percent is set to 10 percent in the example above, and the absolute value of 10 percent is less than the default value of procedure cache size, then procedure cache size will be set to the default value.

With the introduction of dynamically reconfigured memory configuration parameters in Adaptive Server 12.5, an increase in Adaptive Server's memory use will not decrease the size of the procedure cache or the data cache. That is, if your default data cache is set to 40MB, and you increase your procedure cache, the default data cache is still 40MB.

For more information about sp_configure, see the *Reference Manual*.

5   Check the procedure cache requirements.

Stored procedures, triggers, and other compiled objects require more memory to run than older versions. The memory required to run a stored procedure increased by 20 percent between versions 10.x and 11.5. Adaptive Server 12.5 needs approximately 4 percent more procedure cache from version 11.5 for the server to maintain the same performance.

You may increase the procedure cache size during runtime, using sp_configure. You may verify any changes you make to the configuration file without having to restart Adaptive Server using the verify option in sp_configure. The syntax is:

```
sp_configure "configuration file", 0, "verify",
"full_path_to_file"
```

For example:

```
sp_configure "configuration file", 0, "verify",
"/work2/Sybase/ASE125.cfg"
```

For more information about sp_configure and sp_sysmon, see the *Reference Manual* and the *Performance and Tuning Guide*. For more information about configuring memory, see the *System Administration Guide*.

6   Data cache allocation after upgrade. This section is for information only. Adaptive Server ensures that all the data cache sizes after upgrade will be the same as before upgrade.

In pre-12.5 versions of Adaptive Server, the default data cache available depended on the amount of memory available to Adaptive Server. If the default data cache was set to 2MB, Adaptive Server allocated 2MB to the default data cache. If, after memory was allocated to all user-defined data caches and to the procedure cache, there was more memory available, that memory was allocated to the default data cache. However, if there was not enough memory available for Adaptive Server to set up the user-defined caches and the procedure cache, then memory would be taken from the default data cache regardless of the designated configuration parameter. Therefore, it was possible to set default data cache to the default value and have all remaining available memory be allocated to the default data cache.

Adaptive Server 12.5 treats the default data cache size as an absolute value and sets it in the *config* file. The string DEFAULT in the *config* file has a different meaning for default data cache size in Adaptive Server 12.5.

In pre-12.5 Adaptive Server, it meant all the memory left-over after allocating memory for other configurations, procedure and user defined named caches. In Adaptive Server 12.5, it means a default value of 8MB.

During upgrade Adaptive Server ensures that the default data cache size before upgrade is the same as after upgrade. Therefore, during the preupgrade process, the size of the default data cache is obtained and written to the configuration file as an absolute value, not as "DEFAULT." This step is done to enable Adaptive Server 12.5 to have the same default data cache size as before the upgrade. If this size is less than the default size of default data cache of 8MB, then Adaptive Server 12.5 allocates a default data cache of size 8MB.

7 If you unmirrored devices, remirror them, using the disk remirror command.

8 If you used two-phase commit in 11.9.x, run the script to install the two-phase commit tables:

```
isql -Usa -Psa_password -Sserver_name
-i$SYBASE/$SYBASE_ASE/scripts/installcommit
```

You do not need to run this script if you do not use two-phase commit.

Reenable Replication Server

If you disabled replication before the upgrade, you must reenable replication. To do so, first remove any older format log records from your database, and then reenable replication.

Remove old log records

Use the dump database command after the upgrade to dump the database and transaction logs to remove the older format log records from your database. This prevents Replication Server from accessing the pre-upgrade portion of the transaction logs.

For example, to dump the sales database:

```
1> use master
2> go
1> dump database sales to "//./TAPE0"
2> go
```

Reenable replication

Follow the steps in this section to reenable replication after upgrading the primary databases and primary RSSDs.

For each primary database and RSSD:

1   Start Adaptive Server if it is not already running.

2   Log on to the server.

3   Clear the locator for the database by executing the following command in the RSSD for this database:

```
1> use RSSD
2> go
3> rs_zeroltm dataserver, database
4> go
```

4   Enable the truncation point for the database.

```
1> use database
2> go
1> dbcc settrunc("ltm", "valid")
2> go
```

5   Restart the Replication Servers and LTMs.

6   Resume the DSI connections that were suspended before the upgrade by executing the following Replication Server command for each suspended database:

```
1> resume connection to dataserver.database
2> go
```

The replication system is now ready for Adaptive Server 12.5, and applications can resume.

If you installed any Sybase client products, such as Open Client, use the dsedit utility to edit your *interfaces* file and to specify the servers to which you want to connect.

For more information about establishing client/server connections, see the *Open Client Configuration Guide* or Chapter 6, "Installing Sybase PC-Client Products."

Restoring replication after upgrade

Restore the replication functionality:

1   Log into the RSSD for each replicated primary, and for each replicated RSSD, and issue:

```
1> use RSSD_name
2> go

1> rs_zeroltm dataserver, database
2> go
```

2   Log into each replicated primary, and replicated RSSD and issue:

```
1> use database
2> go

1> dbcc settrunc ('ltm', 'valid')
2> go
```

3   Restart the Replication Server.

4   If the database is also used as a RSSD, resume the Replication Server connection to the RSSD by issuing the following command to the Replication Server:

```
1> sysadmin hibernate_off, 'Replication Server'
2> go
```

Make sure to specify the same string that was specified during the 'hibernate_on' command.

5   Log into the Replication Server, and resume the Log Transfer connection for each replicated primary and each replicated RSSD:

```
1> resume log transfer from server.database
2> go
```

If this is a replicated RSSD, you will need to log into the replicate Replication Server.

6   If you are using Rep Agent, log into the Adaptive Server and restart the Rep Agent:

```
1> use database
2> go

1> sp_start_rep_agent database
2> go
```

7   If you are using LTM, restart the LTM.

Reenable auditing

Reenable all auditing options for stored procedures using the audit options you recorded during pre-installation. See "Disable auditing" on page 120. Reenter the audit options using sp_audit.

- Adaptive Server 12.5 includes the following global audit options for sp_audit:

  - security

  - dbcc

- These options are turned off by default and will not be turned on even if you used sp_auditoption all to turn on all global audit options in a version earlier than 11.5.

  To turn on these options, use:

  ```
  1> sp_audit security "on"
  2> go
  1> sp_audit "dbcc" "on"
  2> go
  ```

  For more information, see the *Reference Manual*.

- The functionality provided by these obsolete sp_auditoption options is now covered by the sp_audit security global audit option:

> • server boot
>
> • role toggle

- The security option is turned off by default.

If any of these options were turned off before the upgrade, reset the security option to on to achieve the same auditing actions.

Upgrading Backup Server, Monitor Server, and XP Server

Before you can upgrade Backup or Monitor Server, you must upgrade Adaptive Server.

XP Server does not have a formal upgrade process.

Note Back up all databases, including master, after successfully upgrading Adaptive Server and Backup Server.

❖ **Upgrading Backup and Monitor Servers**

The procedure for upgrading Backup Server and Monitor Server are similar.

- If you are upgrading during the initial installation of Adaptive Server 12.5, select Upgrade Existing Servers from the drop-down menu when the Studio Installer prompts Build or Upgrade ASE, Backup, and Monitor Servers.

    This launches the sqlupgrade utility. Select OK.

- If you are upgrading after the initial installation, start the sqlupgrade utility from the command line. Enter:

        $SYBASE/$SYBASE_ASE/bin/sqlupgrade

4   A warning message appears. Click OK to continue.

5   From the Server selection screen, select the server to upgrade. Click OK.

6   Provide the release path:

- Enter the path where the new Sybase Server is installed.

- Enter the name of component directory. Leave this field blank if the Backup Server is older than 12.0.

- Enter the path to the Open Client/Server Library that the old Backup Server uses. Leave this field blank if it is older than version 12.0.

- Click OK.

7   When you are prompted for the password, enter the password for the SA, and click OK.

8   The Status output screen display the upgrade progress. When the upgrade is complete it displays, "Done." Select OK.

9   You return to the sqlupgrade screen. You can either upgrade another server, or click Exit to leave the utility.

Repeat this procedure to upgrade Monitor Server.

Upgrading compiled objects with *dbcc upgrade_object*

Adaptive Server version 11.9.3 introduced the process of upgrading compiled objects based on their source text. Compiled objects are:

- Check constraints

- Defaults

- Rules

- Stored procedures (including extended stored procedures)

- Triggers

- Views

The source text of each compiled object is stored in the syscomments table, unless it has been manually deleted. When you upgrade the server, the existence of the source text in syscomments is verified during that process. However, the compiled objects are not actually upgraded until they are invoked.

For example, say you have a user-defined stored procedure named list_proc. The presence of source text for list_proc is verified when you upgrade to Adaptive Server 12.5. Then, the first time list_proc is invoked after the upgrade, Adaptive Server detects that the list_proc compiled object has not been upgraded. Adaptive Server recompiles list_proc, based on the source text in syscomments. The newly compiled object is then executed.

Upgraded objects retain the same object ID and permissions that they used prior to being upgraded.

Compiled objects for which the source text was hidden using sp_hidetext are upgraded like objects for which the source text is not hidden. For information on sp_hidetext, see the *Reference Manual*.

Note If you are upgrading from 32-bit installations to use a 64-bit Adaptive Server, the size of each 64-bit compiled object in the sysprocedures table in each database increases by approximately 55 percent when the object is upgraded. The pre-upgrade calculates the exact size. Increase your upgraded database size accordingly.

To ensure that compiled objects have been upgraded successfully *before* they are invoked, you can upgrade them manually using the dbcc upgrade_object command. For details, see "Finding compiled object errors before production" on page 140.

Finding compiled object errors before production

Changes made in earlier versions of Adaptive Server may cause compiled objects to work differently in version 12.5 and later. You can use dbcc upgrade_object to find the following errors and potential problem areas that may require manual changes to achieve the correct behavior:

- Reserved word errors
- Missing, truncated, or corrupted source text
- Quoted identifier errors
- Temporary table references
- select * potential problem areas

After reviewing the errors and potential problem areas, and fixing those that need to be changed, you can use dbcc upgrade_object to upgrade compiled objects manually instead of waiting for the server to upgrade the objects automatically. For details, see "Using dbcc upgrade_object" on page 143.

Reserved word errors

If dbcc upgrade_object finds a reserved word used as an object name in a compiled object, it returns an error, and the upgrade of that object fails. To fix the error, either manually change the object name or use quotes around the object name and issue the command set quoted identifiers on. Then, drop and re-create the compiled object.

For example, suppose you load a database dump from Adaptive Server 11.5 into Adaptive Server 12.5 and the dump contains a stored procedure that uses the word "lock." When you run dbcc upgrade_object on that stored procedure, the command returns an error because, although "lock" was not reserved in version 11.5, it became a reserved word in version 11.9.2. With this advance notice, you can change the stored procedure and any related tables before they are used in a production environment.

Missing, truncated, or corrupted source text

If the source text in syscomments was deleted, truncated, or otherwise corrupted, dbcc upgrade_object may report syntax errors. If the source text was not hidden, you can use sp_helptext to verify the completeness of the source text. If truncation or other corruption has occurred, drop and re-create the compiled object.

Quoted identifier errors

dbcc upgrade_object returns a quoted identifier error if:

- The compiled object was created in a pre-11.9.2 version with quoted identifiers active (set quoted identifiers on).

- Quoted identifiers are not active (set quoted identifiers off) in the current database.

To avoid this error, activate quoted identifiers before running dbcc upgrade_object. When quoted identifiers are active, you must use single quotes instead of double quotes around quoted dbcc upgrade_object keywords.

If quoted identifier errors occur, use the set command to activate quoted identifiers, and then run dbcc upgrade_object to upgrade the object.

For compiled objects created in 11.9.2 or later, the upgrade process automatically activates or deactivates quoted identifiers as appropriate.

Note Quoted identifiers are not the same as literals enclosed in double quotes. The latter do not require you to perform any special action before the upgrade.

Temporary table references

If a compiled object such as a stored procedure or trigger refers to a temporary table (#temp *table_name*) that was created outside the body of the object, the upgrade fails, and dbcc upgrade_object returns an error. To correct this error, create the temporary table exactly as expected by the compiled object, and then execute dbcc upgrade_object again. This is not necessary if the compiled object is upgraded automatically when it is invoked.

select * potential problem areas

In Adaptive Server version 11.9.3 and later, the results of a select * clause in a stored procedure, trigger, or view that was created in an earlier version of Adaptive Server may be different from what you expect.

For more information about the changes, see the *Reference Manual*.

If dbcc upgrade_object finds a select * clause in the outermost query block of a stored procedure, it returns an error, and does not upgrade the object.

For example, consider the following stored procedures:

```
create procedure myproc as
  select * from employees
go
create procedure yourproc as
  if exists (select * from employees)
    print "Found one!"
go
```

dbcc upgrade_object returns an error on myproc because myproc includes a statement with a select * clause in the outermost query block. This procedure is not upgraded.

dbcc upgrade_object does not return an error on yourproc because the select * clause occurs in a subquery. This procedure is upgraded.

Determining whether *select* * should be changed in views	If dbcc upgrade_object reports the existence of select * in a view, compare the output of syscolumns for the original view to the output of the table, to determine whether columns have been added to or deleted from the table since the view was created.

For example, suppose you have the following statement:

```
create view all_emps as select * from employees
```

Before upgrading the all_emps view, use the following queries to determine the number of columns in the original view and the number of columns in the updated table:

```
select name from syscolumns
  where id = object_id("all_emps")
select name from syscolumns
  where id = object_id("employees")
```

Compare the output of the two queries. If the table contains more columns than the view, and retaining the pre-upgrade results of the select * statement is important, change the select * statement to a select statement with specific column names. If the view was created from multiple tables, check the columns in all tables that comprise the view and rewrite the select statement if necessary.

Warning! Do not execute a select * statement from the view. Doing so upgrades the view and overwrites the information about the original column information in syscolumns.

Another way to determine the difference between the columns in the view and in the new tables is to run sp_help on both the view and the tables that comprise the view.

This comparison works only for views, not for other compiled objects. To determine whether select * statements in other compiled objects need to be revised, review the source text of each compiled object.

Using *dbcc upgrade_object*

Syntax	dbcc upgrade_object [(*dbid* | *dbname* [, ['*database.*[*owner*].]*compiled_object_name*' | 'check' | 'default' | 'procedure' | 'rule' | 'trigger' | 'view' [, 'force']])]

where:

- *dbid* specifies the database ID. If you do not specify *dbid*, all compiled objects in the current database are upgraded.

- *dbname* specifies the database name. If you do not specify *dbname*, all compiled objects in the current database are upgraded.

- *compiled_object_name* is the name of a specific compiled object you want to upgrade. If you use the fully qualified name, *dbname* and *database* must match, and you must enclose the fully qualified name in quotes. If the database contains more than one compiled object of the same name, use the fully qualified name. Otherwise, all objects with the same name are parsed, and if no errors are found, upgraded.

- check upgrades all check constraints and rules. Referential constraints are not compiled objects and do not require upgrading.

- default upgrades all declarative defaults and the defaults created with the create default command.

- procedure upgrades all stored procedures.

- rule upgrades all rules and check constraints.

- trigger upgrades all triggers.

- view upgrades all views.

  The keywords check, default, procedure, rule, trigger, and view specify the classes of compiled objects to be upgraded. When you specify a class, all objects in that class, in the specified database, are upgraded, provided that dbcc upgrade_object finds no errors or potential problem areas.

- force specifies that you want to upgrade the specified object even if it contains a select * clause. Do not use force unless you have confirmed that the select * statement will not return unexpected results. The force option does not upgrade objects that contain reserved words, contain truncated or missing source text, refer to nonexistent temporary tables, or do not match the quoted identifier setting. These objects must be fixed before they can be upgraded.

Note If set quoted identifiers is on, use single quotes around the keywords. If set quoted identifiers is off, you can use either double quotes or single quotes.

Examples

```
dbcc upgrade_object
```

Upgrades all compiled objects in the active database.

```
dbcc upgrade_object(listdb, 'procedure')
```

144

Upgrades all stored procedures in the listdb database. Single quotes are used around procedure because set quoted identifiers is on.

```
dbcc upgrade_object(listdb, "rule")
```

Upgrades all rules and check constraints in the listdb database. Double quotes are used around rule because set quoted identifiers is off.

```
dbcc upgrade_object(listdb, list_proc)
```

Upgrades all stored procedures named list_proc in the listdb database.

```
dbcc upgrade_object(listdb,
    "listdb.jkarrik.list_proc")
```

Upgrades the stored procedure list_proc, which is owned by the login "jkarrik".

```
dbcc upgrade_object(master,
    "listdb.jkarrik.list_proc")
```

Returns an error because the value of *dbname* is master and the value of *database* is listdb. These values must match.

Permissions   Only the Database Owner or a System Administrator can execute dbcc upgrade_object. The Database Owner can upgrade his or her own objects in the database.

Upgraded objects retain the same owner that they had prior to being upgraded.

Increasing the log segment size

You can specify that all compiled objects of a particular class should be upgraded in one execution of dbcc upgrade_object; for example, you can upgrade all triggers by using the trigger keyword. However, even though you use only one dbcc command, the upgrade of each object is recorded in a separate transaction; the old row is deleted from sysprocedures and a new row is written. Therefore, if you run dbcc upgrade_object on a large number of compiled objects, your system may run out of log space. Increase the size of the log segment in the databases in which you plan to run this command, to allow sufficient room to log all the upgrades.

Error reporting

To send all the output from dbcc upgrade_object to the screen, a System Administrator can start the server with dbcc traceon(3604). Sybase recommends that you use this command if you think the output of error messages might overflow the error log.

Using database dumps in upgrades

Upgrading using dump and load

You can load pre-12.5 database dumps and transaction logs and upgrade the databases. To determine the size requirements for the upgrade, see the discussion about size requirements in "sybsystemprocs" on page 115.

Some issues of which you should be aware:

- Upgrade requires space for copying data and logging changes to the system tables during upgrade. If the source database in the dump was nearly full, the upgrade process might fail due to insufficient space. While this is expected to be uncommon, you can use alter database to extend the free space in the event of insufficient-space errors.

- After reloading an older dump, run sp_checkreswords from the new installation on the loaded database to check for reserved words.

Upgrading compiled objects in database dumps

When you load a database dump that was created in an earlier version than the current Adaptive Server, you are not required to perform the pre-upgrade tasks before loading the dump. Therefore, you will not receive any notification if the compiled objects in your database dump are missing their source text. After loading a database dump, run sp_checksource to verify the existence of the source text for all compiled objects in the database. Then, you can allow the compiled objects to be upgraded as they are executed, or you can run dbcc upgrade_object to find potential problems and upgrade objects manually.

For information on using sp_checksource, see the *Reference Manual*.

Determining whether a compiled object has been upgraded

To determine whether a compiled object has been upgraded, do one of the following:

- Look at the sysprocedures.version column. If the object was upgraded, this column will contain the number 12500.

- If you are upgrading to a 64-bit pointer size in the same version, look at the sysprocedures.status column. It will contain a hexadecimal bit setting of 0x2 to indicate that the object uses 64-bit pointers. If the bit is not set, the object is a 32-bit object, which means it was not upgraded.

Migrating from 32-bit to 64-bit versions

Adaptive Server sqlupgrade only upgrades Adaptive Server from one version to another. This section describes how to upgrade a 32-bit version of Adaptive Server to a 64-bit version of Adaptive Server.

To migrate Adaptive Server from the 32-bit to the 64-bit version, you must first install and configure the 64-bit operating system.

Method #1: Dump and load

1   Run dbcc checks on all databases in the 32-bit Adaptive Server and make sure they run cleanly.

2   Create a 64-bit Adaptive Server in a new directory

3   Create devices and databases to match those in the 32-bit server. Make sure that the sysusages mapping is correct. See document ID #1324 at http://my.sybase.com/detail?id=1324.

> **Note** Allow 10 percent more space for the sybsystemprocs database.

4   Dump the databases on the 32-bit server.

5   Load the databases to the 64-bit server.

6   If you have partitioned tables, update partition statistics.

7   Run dbcc checks on the 64-bit server and make sure they run cleanly.

Method #2: BCP data out and in

If you have DDL scripts to create devices, databases, tables, rules, stored procedures, triggers, and views you can use bcp to move data out of the old database and into the new.

If you do not have DDL scripts, use the ddlgen utility to re-create the schema for the Adaptive Server you are about to upgrade. See the *Utility Guide*.

1   Run dbcc checks on databases to verify data integrity.

2   Use bcp to extract all the data from all the tables in the databases.

3   Create a new 64-bit Adaptive Server in a new directory.

4   Create devices, databases, and tables.

5   bcp data into tables.

6   Recreate all views, triggers, stored procedures.

7   Run dbcc checks on the 64-bit server and make sure they run cleanly.

Method #3: Replacing the binary

The third method requires you to modify some key files, recreate stored procedures, then replace the binary.

1   Run dbcc checks on all databases in the 32-bit Adaptive Server and make sure the dbcc's run cleanly. (checkdb, checkalloc, checkcatalog and checkstorage)

2   In a new directory, copy the files for the 64-bit Adaptive Server.

3   Shut down the 32-bit server.

4   Copy the *interfaces* file and the configuration file from the 32-bit *$SYBASE* directory to the 64-bit *$SYBASE* directory.

5   Copy the 32-bit *$SYBASE/install RUN_server* file to the equivalent 64-bit *$SYBASE/install* directory.

6   Edit the *RUN_server* file to reflect the new location of the *interfaces*, configuration, and log files.

7   Remove all references to the 32-bit *$SYBASE* directories from your *$PATH* definition.

8   cd to the 64-bit *$SYBASE* and source the *SYBASE.csh* script (c shell) or *SYBASE.sh* script (korn shell).

9   cd to the 64-bit *$SYBASE/install* and execute:

```
startserver -f RUN_server
```

10   After the 64-bit server starts, run installmaster, installmodel, and installmsg.ebf.

11   If you used dbccdb for dbcc checkstorage, run the *installdbccdb*. This recreates some tables in the dbccdb, and you may loose history data.

12   Drop and recreate all your compiled objects, such as stored procedures, triggers, views, and defaults.

13   If you have partitioned tables, update partition statistics.

14   Run dbcc again on all databases to verify that they run cleanly.

Recovering from a failed upgrade

•   A common failure you might encounter, even after going through the pre-upgrade tasks, is that the log may get full due to the catalog changes that are being done during upgraded. If so, log in via isql to the new server, and issue:

```
isql> dump tran dbname with no_log
```

This will free up the log space, and allow the upgrade process to continue.

• In some cases, the space estimations done by preupgrade might be insufficient for the data copy phase of upgrade. In that case, you may get an error that there is insufficient space in the system segment for the upgrade step. The upgrade process will hang, waiting for space to be provided. You can log in to the new server using isql, and use alter database to increase the size of the database.

```
isql> alter database dbname on device_name = "2m"
```

Note In 12.5 Adaptive Server, alter database allows you to specify the size to alter the database with the unit-specifier 'm', or 'M'.

CHAPTER 9   **Removing Sybase Servers**

Uninstalling Sybase Servers

1   Stop the Monitor Server and the Backup Server before stopping Adaptive Server. See Chapter 7, "Starting and Stopping Servers."

2   If you have not already done so, source the SYBASE script from the *$SYBASE* directory. For a C shell, enter:

```
source SYBASE.csh
```

For a Bourne shell, enter:

```
. SYBASE.sh
```

3   Start the Studio Installer. From the *$SYBASE* directory, run:

```
    Installer/bin/uninstall
```

> **Note** If you do not have a "." (dot) in the PATH, you must add "./" before the script at the command line.

This starts the Studio Installer and launches the Uninstall window.

4   The Studio Installer prompts: "Uninstall components now?"

Click Yes.

5   A component selection window appears. Select individual components by clicking the check box to the left of the component, or click Select All to remove all components.

Click Next.

6   The Uninstall summary window displays the components selected for removal.

Click Next.

The Studio Installer begins removing the selected components from the system. A progress indicator notifies you when the components have been successfully removed.

7   Click OK to exit the application.

The Studio Installer removes:

• All files installed from the CD

• All directories it created, unless the directory is not empty

After running the uninstall utility, a message displays stating that you should remove the remaining directories and files. Remove these directories and files after you finish running the utility.

Removing an existing Adaptive Server

1 To remove an existing Adaptive Server, from a command-line prompt enter:

```
rm servername.*
```

2 Change to the directory *$SYBASE/$SYBASE_ASE/install* and run the following commands:

```
rm RUN_servername.*
```

```
rm servername.*
```

3 Edit the *interfaces* file, *$SYBASE/interfaces* to remove all references to the Adaptive Server.

4 If you used operating system files for database devices, remove those.

Troubleshooting

This chapter provides instructions for troubleshooting installation error messages.

If this chapter does not describe the error message you are encountering, see the *Error Messages and Troubleshooting Guide*.

Topic	Page
Overview	153
Error log locations	154
Solutions to common installation problems	155
Troubleshooting SySAM	162

Overview

For Sybase server products, there are two categories of errors:

*   Errors generated by the installation, upgrade, and setup utilities

*   Errors generated by the server (Adaptive Server, Backup Server, and so on)

To determine the cause of an error, first look in the log file of the utility being used, to identify the task the utility was performing when it failed. Then check the server error log. See "Error log locations" on page 154 for the location of the error log files for the installation utilities and the servers.

Table 10-1 lists possible causes and solutions for common problems that you might encounter during a first-time installation or upgrade. If you continue to have problems, retry the installation or upgrade.

If the installation program or svrbuild unexpectedly quits, or if you cannot correct the problem, see the *Error Messages and Troubleshooting Guide*.

Table 10-1: Troubleshooting guidelines

Problem	Possible cause and solution
The installation program cannot start Adaptive Server.	Failure to start Adaptive Server is generally caused by a lack of available RAM or disk space.
	Make sure you meet RAM requirements. If you have the required RAM, remove, then reinstall all applications to the hard drive and restart the installation process.
	After Adaptive Server is installed, there should be 25MB of free disk space left in the disk drive. Adaptive Server needs approximately 18MB for creating shared memory files.
	Verify that you are logged in as System Administrator. You must log in as an administrator to start Adaptive Server.
	Shut down Monitor Server before restarting Adaptive Server.
After upgrading Adaptive Server, you cannot use svrbuild.	After you begin upgrading a server, you may be unable to use the same svrbuild session for other tasks. Exit and restart svrbuild.
The installation program cannot connect to the upgraded Adaptive Server.	After you begin upgrading a server, you may be unable to use the same svrbuild session for other tasks. Exit and restart svrbuild.
The installation program detects reserved word conflicts.	"Stopping Adaptive Server after a failure" on page 158.
The upgrade fails.	See "Recovering from a failed upgrade" on page 160.

Error log locations

The information in the error logs can help you determine the reason and possible solution for an error message.

For installation utilities

Table 10-2 lists the default error log locations for the installation, upgrade, and setup utilities.

Table 10-2: Error log locations for installation utilities

Utility	Error log location
Studio Installer	*$SYBASE/installer.log*
srvbuild srvbuildres	*$SYBASE/$SYBASE_ASE/init/logs/srvbuildMMDD.NNN* where *MM* is the month, *DD* is the date, and *NNN* is a three-digit number identifying the srvbuild session
sqlloc sqllocres	*$SYBASE/$SYBASE_ASE/init/logs/sqllocMMDD.NNN*
sqlupgrade sqlupgraderes	• *$SYBASE/$SYBASE_ASE/init/logs/sqlupgradeMMDD.NNN* • *$SYBASE/$SYBASE_ASE/upgrade/upgrade.NNN* (a temporary file created by the upgrade process)

For Sybase servers

Table 10-3 lists the default error log locations for each Sybase server.

Table 10-3: Error log locations for Sybase servers

Server	Default error log path and file name
Adaptive Server	*$SYBASE/$SYBASE_ASE/install/servername.log*
Backup Server	*$SYBASE/$SYBASE_ASE/install/servername_back.log*
Monitor Server	The directory from which Monitor Server is started. The error log file name is *ms.log*.
XP Server	Writes to the Adaptive Server error log.

Solutions to common installation problems

If this section does not describe the problem you are experiencing, see the *Error Messages and Troubleshooting Guide*.

Unable to use X-Windows

If the setup and installation utilities do not display correctly, you may have to adjust the resolution on your monitor.

To change to a smaller font size issue the following UNIX commands:

```
% cd $SYBASE
% chmod +w xappdefaults
% cd xappdefaults
% chmod +w *
% foreach i(*)
? cat $i | sed -e "s/140/100/g" | sed -e "s/^#D/D/g" | sed -e "s/^#S/S/g" > p
? mv p $i
? end
%
```

The installation utilities will use approximately 25 percent less screen space.

Fatal Error: Cannot Map *libct.so*

If you execute srvbuild and get this message:

```
22900:srvbuild: /sbin/loader: Fatal Error: cannot map
libct.so
```

You need to set the shared library environment variable. For more information on setting this environment variable, see Chapter 5, "Post-Installation Tasks."

Note Depending upon your platform, the file names and the error message may vary slightly.

Cannot eject the CD from the drive

If you cannot eject the CD from the drive:

• Check to see whether the CD drive path is the current directory (pwd) in a UNIX terminal window. If it is, change (cd) to another directory.

• Check for sybhelp processes. If these processes exist, kill them using the UNIX kill command.

DISPLAY environment variable not set correctly

If you run Studio Installer and you get this error message:

```
The DISPLAY environment variable is not set correctly.
```

it means that the DISPLAY environment variable on the *remote* machine is not set correctly to display the Studio Installer user interface to your *local* machine.

To correct the problem, enter the following command at the UNIX prompt of the *remote* machine:

For C shell:

```
setenv DISPLAY host_name:0.0
```

For Bourne shell:

```
DISPLAY=host_name:0.0; export DISPLAY
```

where *host_name* is the name of the machine on which you want the Studio Installer interface to appear (that is, on your *local* machine).

Client not authorized to connect to server

If you run Studio Installer and you get this error message:

```
Xlib: connection to "host_name" refused by server
Xlib: Client is not authorized to connect to Server
xhost: unable to open display "host_name"
```

it means the *remote* machine does not have permissions to display the user interface on the *local* machine where you are working.

To correct the problem, enter the following command at the UNIX prompt of your *local* machine:

```
xhost +remote_machine
```

where *remote_machine* is the machine on which you are running Studio Installer.

Restart Studio Installer.

Address already in use

If you receive the following message from srvbuild:

```
kernel: ninit: bind, Address already in use
```

it means you entered a port number that is already in use.

To correct the problem, enter a different port number on the srvbuild screen. The command netstat -a produces a list of port numbers in use.

Adaptive Server failed to start

The shared memory of the operating system may not be set high enough.

To correct the problem, see Chapter 4, "Installing Sybase Servers" for instructions on adjusting the shared memory value and restart the installation or upgrade process.

Cannot start XP Server

You may receive the following message from XP Server, when it is invoked by xp_cmdshell or some other extended stored procedure:

```
Msg 11018, Level 16, State 1:
Procedure "xp_cmdshell", Line 2:
XP Server must be up for ESP to execute.
(return status = -6)
```

Verify that there is an XP Server entry in Adaptive Server's sysservers table. If you created XP Server separately (in a different srvbuild session) from Adaptive Server and you did not specify a related Adaptive Server, srvbuild cannot update the sysservers table.

Use sp_addserver to add an entry to the sysservers table.

Stopping Adaptive Server after a failure

If the installation or upgrade session fails after you start Adaptive Server, use the shutdown command:

1   Log on as "sa".

2   Shut down Adaptive Server using the shutdown with nowait command. Using the with nowait option stops the Adaptive Server immediately, without waiting for currently executing SQL statements to finish:

```
1> shutdown with nowait
2> go
```

Recovering from a failed installation

If the installation does not succeed, the installation program displays error messages. Review the error messages and your Adaptive Server error log to determine the cause of the installation failure. For default error log locations, see Table 10-3.

If installation fails after files are created

If the installation program quits while you are configuring Adaptive Server, perform the following steps:

1   View the contents of the log file generated by Adaptive Server. For default error log locations, see Table 10-3.

2   Take any suggested actions to correct the problem.

3   If the installation fails *after* the installation program has created any operating system files, such as the master device or system procedures device files, delete those files.

4   If the installation fails *after* the installation program starts the Adaptive Server that you are attempting to install, shut down that server. Follow the procedure under "Stopping Adaptive Server" on page 97 before performing step 5.

5   Use asecfg to restart the configuration.

Troubleshooting resource file installations

If you encounter problems during the build, configure, or upgrade of Adaptive Server using the srvbuild[res], sqlloc[res], or sqlupgrade[res], it may be that these utilities did not allow enough time for Adaptive Server to shut down properly.

You can set the environment variable SYBSHUTWAIT to force the utilities to wait for Adaptive Server to shutdown. For example:

```
% setenv SYBSHUTWAIT 120
```

This forces the utility to wait for 2 minutes to allow Adaptive Server to shut down before proceed with the next task.

If Adaptive Server or SQL Server fails the pre-upgrade test, sqlupgrade displays:

```
Server SERVER_NAME failed preupgrade eligibility test.
See log for more information.
```

159

1 From the Upgrade window, select Exit.

2 Examine the log file created in the *$SYBASE/$SYBASE_ASE/init/logs* directory to find out why Adaptive Server or SQL Server failed the pre-upgrade eligibility test.

If the log contains messages about insufficient space in sybsystemprocs, follow the instructions in "Increasing the size of the sybsystemprocs database" on page 116 to correct the problem.

After you resolve any problems, shut down Adaptive Server or SQL Server and use sqlupgrade to complete the upgrade session.

Recovering from a failed upgrade

If the upgrade process does not succeed, the installation program displays error messages. Review the error messages and the Adaptive Server error log to determine the cause of the upgrade failure. For default error log locations, see Table 10-2.

Restoring from backup

You may need to restore your databases due to a failed upgrade.

If you think the upgrade failure or its cause may have damaged your databases, restore the databases from backups. For information on restoring databases, see the *System Administration Guide*.

If you are concerned about the possible corruption of your databases, exit sqlupgrade, but do not attempt to restart the upgrade session until you have restored the databases from backup. After restoration is complete, retry the upgrade.

Rerunning the upgrade

Whether you can safely re-run the upgrade depends on when the failure occurred in the upgrade process. If the failure occurs while the message: "Starting to upgrade Adaptive Server" is displayed, it is safe to re-run the upgrade program.

1 Try to fix the problem that caused the upgrade to fail.

2 Run the upgrade again.

If the upgrade fails:

- *Before* returning the message "Setting upgrade version to 12.5", you may need to restore your latest database backup, and restart the upgrade.

- *After* returning the message "Setting upgrade version to 12.5", it is not necessary to restart the upgrade. The installation utility considers the upgrade to be complete.

Also, it is not necessary to restore a database from a backup unless that database failed during the upgrade.

Recording the upgrade manually

If Adaptive Server did not finish recording the upgrade in the sysattributes table before the failure occurred.

1   Fix the problem that caused the failure.

The first error message indicates the cause of the failure. If you can, solve the problem and proceed to step 2. For example, you can usually correct an 1105 error with a dump transaction command. However, more complex problems may have to be referred to Sybase Technical Support.

2   Execute the following SQL statements to allow Adaptive Server to complete recording the upgrade:

```
1> declare @dbname varchar(30)
2> select @dbname = min(name)
3> from sysdatabases
4> while @dbname is not null
5> begin
6> online database @dbname
7> select @dbname = min(name)
8> from sysdatabases
9> where name > @dbname
10> end
```

If the cause of the failure is known

If the error logs or messages clearly indicate the cause of failure, and you do not believe your databases were damaged, you can attempt to fix the problem and re-run the upgrade immediately.

1   Exit the sqlupgrade program.

2   Perform the necessary actions to fix the problem.

For example, if the error log indicates that the upgrade failed because your existing databases do not contain enough space, use the alter database command to increase the available space.

3   It may be necessary to shut down Adaptive Server. Follow the instructions for "Stopping Adaptive Server" on page 97.

Shutting down the server enables the installation program to start the server and re-run the upgrade session.

4   Start sqlupgrade again.

5   Select Upgrade Adaptive Server, and proceed with the upgrade.

If the cause of the failure is unknown

If the upgrade fails again, and you cannot determine the cause of failure, check the error log file to find out when and where the upgrade failed, and contact Sybase Technical Support.

By default, the log file is located in *$SYBASE/$SYBASE_ASE/install/errorlog*.

Troubleshooting SySAM

When you start Adaptive Server with SySAM support, problems acquiring licenses or contacting the asset management software appear in the Adaptive Server error log file, *lmgrd.log* (*$SYBASE/$SYBASE_SYSAM/log*).

Authorization code input error

When you purchase licenses for Sybase Adaptive Server products, you are issued a Sybase Software Asset Management Certificate. The certificate has the following information for each product:

• Order Number

• Feature Name

• Feature Count

• Software Version

• Authorization Code

- Product Description

This information is used by SySAM to build the license file, with new licensed features appended to the end of the file. Here is a sample license file, *license.dat*:

```
SERVER server1 ANY 4100
VENDOR SYBASE $SYBASE/$SYBASE_SYSAM/bin/SYBASE

USE_SERVER

INCREMENT ASE_SERVER SYBASE 12.5 PERMANENT 1000
123456789123 SN=10001 OVERDRAFT=10000 ck=0

INCREMENT ASE_JAVA SYBASE 12.5 PERMANENT 1000
123456789123 SN=10001 OVERDRAFT=10000 ck=0

INCREMENT ASE_DTM SYBASE 12.5 PERMANENT 1000
123456789123 SN=10001 OVERDRAFT=10000 ck=0

INCREMENT ASE_HA SYBASE 12.5 PERMANENT 1000
123456789123 SN=10001 OVERDRAFT=10000 ck=0
```

- *ASE_<FEATURE>* is the feature name, such as ASE_SERVER, ASE_JAVA, and so on.

- *12.5* is the version number.

- *Feature Count* immediately follows the license type, *PERMANENT*.

- *SN=10001* is the Order Number.

- *OVERDRAFT= ###* is the maximum licenses that can be checked out.

- *123456789123* is a 12-digit number representing the authorization code.

The authorization code is case sensitive. If you make a mistake while entering the authorization code, correct it by accessing the license file with a text editor, making the necessary changes, and saving the file.

The file is located in *$SYBASE/$SYBASE_SYSAM/licenses/license.dat*.

Warning! Tampering with any portion of the licenses file other than the authorization code invalidates the license.

Alternative Installation Methods

Resource file installation

You can create an Adaptive Server or Backup Server, using values specified in a resource file that defines the attributes for the server.

Resource files are ASCII format template files that contain configuration variables. To use the template files, edit the resource file, replace variables with desired values, and execute the svrbuildres utility. The svrbuildres utility uses the variables to create servers.

Resource files allow you to create servers in batch mode. You can create servers on multiple systems from the same resource file by editing a few values in that file. Within a single resource file, you can create only one server at a time.

The server files must already exist on the computer before the servers can be created. Use procedures in Chapter 4, "Installing Sybase Servers."

Note You cannot install Monitor Server, XP Server or the full-text search engine using resource files.

To install an Adaptive Server or Backup Server using resource files:

1   Edit a resource file as described in "Installing from a resource file" on page 165.

2   Execute the srvbuildres utility, using the edited resource file as described in "Using srvbuildres" on page 168.

Installing from a resource file

You can install Sybase servers via a resource file. You can use either:

•   The sample resource file provided with the software, or

•   The resource file that is created by the srvbuild utility.

Editing a sample resource file

Sample resource files for creating Adaptive Server and Backup Server are included in your Adaptive Server distribution in:

$SYBASE/$SYBASE_ASE/init/sample_resource_files.

Edit these files to specify the attributes for the servers you want to create.

Note the following:

•   If you use USE-DEFAULT as the Adaptive Server name, the utility substitutes the name of the computer on which it is running for the server name. If you use USE-DEFAULT, do not create Backup Server entries in the resource file. They are created automatically using the default name.

•   To create a server with a different name, change the Adaptive Server name and use the new name with the "_BK" extensions for Backup Server. For example, the Backup Server for PIANO should be PIANO_BK.

•   The master device size must be at least 30MB.

•   The system procedure device and sybsystemprocs database sizes must be at least 100MB.

•   The auditing entries in the resource file are not supported and will not be processed by the utility. To enable auditing, see *Configuring Adaptive Server Enterprise*.

Editing a resource file created by *srvbuild*

You can create a resource file by running the srvbuild utility. A new file containing the values you specified for the Adaptive Server or Backup Server is written to:

$SYBASE/$SYBASE_ASE/init/logs/srvbuildMMDD.VVV-servername.rs.

Edit these files to specify the attributes for the servers you want to create.

Warning! Be sure to edit the resource file generated by srvbuild to change attributes, such as device names, that may be different on another system.

Resource file attributes for Adaptive Server

Table A-1 shows the Adaptive Server resource file attributes, their default values, and other options.

The attributes in bold type are *required*.

All alpha values are case sensitive.

The prefix of the attribute name varies, depending on whether the resource file was created by srvbuild or by sybinit (from a pre-11.5 SQL Server). The prefix is ignored by srvbuildres when processing the resource file.

The attribute names in Table A-1 are the attribute names provided in the sample resource file included in your Adaptive Server distribution.

Table A-1: Resource file attributes for Adaptive Server

Attribute	Default value [other options]
sybinit.release_directory	The value of *$SYBASE* at your site
sybinit.product	sqlsrv
sqlsrv.server_name	*server_name*
sqlsrv.new_config	yes (required value)
sqlsrv.do_add_server (to *interfaces* file)	yes [no]
sqlsrv.network_protocol_list	tcp [spx]
sqlsrv.network_hostname_list	*hostname*
sqlsrv.network_port_list	*port_number*
sqlsrv.master_device_physical_name	*path_and_name_of_master_device*
sqlsrv.master_device_size	30 (MB)
sqlsrv.master_database_size	5 (MB)
sqlsrv.errorlog	*$SYBASE/$SYBASE_ASE/install/server_name.log*
sqlsrv.do_upgrade	no (required value)
sqlsrv.sybsystemprocs_device_physical_name	*path_and_name_of_sybsystemprocs_device*
sqlsrv.sybsystemprocs_device_size	100 (MB)
sqlsrv.sybsystemprocs_database_size	100 (MB)
sqlsrv.sybsytemdb_device_physical_name	*path_and_name_of_ sybsystemdb_device*
sqlsrv.sybsystemdb_device_size	5 (MB)
sqlsrv.sybsystemdb_database_size	5 (MB)
sqlsrv.default_backup_server	*server_name*_back

Resource file attributes for Backup Server

Table A-2 shows the Backup Server resource file attributes, their default values, and other options.

The attributes in bold type are *required*.

All values are case sensitive.

The prefix of the attribute name varies, depending on whether the resource file was created by srvbuild or by sybinit (from a pre-11.5 SQL Server). The prefix is ignored by srvbuildres when processing the resource file.

The attribute names in Table A-2 are the attribute names provided in the sample resource file included in your Adaptive Server distribution.

Table A-2: Resource file attributes for Backup Server

Attribute	Default value [other options]
sybinit.release_directory	$SYBASE
sybinit.product	bsrv
bsrv.server_name	*server_name*_back
bsrv.do_add_backup_server (to *interfaces* file)	yes [no]
bsrv.network_protocol_list	tcp [spx]
bserv.network_hostname_list	*hostname*
bsrv.network_port_list	*port_number*
bsrv.language	us_english [chinese, french, german, japanese, spanish]
bsrv.character_set	roman8 [iso_1, cp850, cp437, deckanji, ascii_8, eucgb, eucjis, mac, roman8, sjis, utf8]
bsrv.tape_config_file	*$SYBASE/$SYBASE_ASE/ backup_tape.cfg*
bsrv.errorlog	*$SYBASE/$SYBASE_ASE/ server_name_back.log*

Using *srvbuildres*

To execute srvbuildres at the UNIX prompt enter:

```
$SYBASE/$SYBASE_ASE/bin/srvbuildres -r
resource_file
```

where *resource_file* specifies the resource file containing the attributes that describe the server to build.

Note Unlike srvbuild, the srvbuildres utility does not require any X libraries.

When you have completed resource file installation, go to Chapter 5, "Post-Installation Tasks."

Installing files in NOGUI mode

To install components in a NOGUI environment, perform all pre-installation preparation steps as described; however, replace the window-based Studio Installer instructions (Chapter 4, "Installing Sybase Servers") with these instructions.

There are several arguments you can use when you run Studio Installer from the command line. The syntax is:

```
install -argument
```

*   -debug:

    Use this argument to display output normally sent to the *Installer.log*.

*   -f *filename*

    Use this argument to provide the file name of the file that contains all the input you want to use for a silent installation.

    Before using this argument, you must create a file containing all the installation inputs (target directory, components to be installed, and so on).

*   -c

    Use this argument run the installer without the graphical user interface. The Studio Installer prompts the user for the required user input.

*   -trace

Use this argument to print more detailed information to the
Installer.log, which is useful for tracking abnormal executions of
installations.

• -version

Use this argument to obtain version information for the installer. Do
not use any other arguments when using this argument.

Complete the following steps:

1   Mount the CD-ROM:

```
/etc/mount -F cdfs -o ro device_name /cdrom
```

where *device_name* is the CD-ROM drive device name and */cdrom* is
the name of the directory where the CD is to be mounted.

Note The location of the mount command is site-specific, and might
be different than what is shown in these instructions. If you cannot
mount the CD-ROM drive using the path shown, consult your
operating system documentation or contact your operating system
administrator.

---

2   At the prompt, type:

```
cd /CDROM
./install -c
```

3   Select the type of install to perform.

• Full Install

• Standard Install

• Cancel

4   From the prompt, enter the installation directory (the default is
/usr/sybase/).

5   Select one of the following:

• Continue – to continue.

• Previous Menu – to return to the previous screen.

• Cancel – to terminate the installation procedure.

6   The installer prompts: "Have you registered Sybase Software Asset
Management Certificates at a central license host?"

If Yes, enter the following information about the primary network node from which this machine will be checking out licenses:

- License Manager Host is the host name of the primary network node on which you installed the asset management software.

- License Manager Port is an unused port number on the primary network node. The local Adaptive Server uses this port number to contact the primary network node when checking out licenses.

For information on licensing features on the primary license host, see Chapter 3, "Sybase Software Asset Management (SySAM)."

Completing the installation process

1   Follow the directions on the screen.

2   When the installation is complete, use the asecfg utility to:

- Configure a New Server – launches the srvbuild utility (Chapter 4, "Installing Sybase Servers.")

- Upgrade an Existing Server – launches the sqlupgrade utility (Chapter 8, "Upgrading Sybase Servers.")

- Localize an Existing Server – launches the sqlloc utility. See *Configuring Adaptive Server Enterprise*.

- Edit the Interface File – launches the dsedit utility to modify the *interfaces* file.

- Exit – exits the configuration utility.

    For information about these Sybase installation and configuration utilities, see the *Utility Guide* for your platform.

3   Exit the asecfg utility.

4   Check the installation log file to see a record of the installation process. The server log file is located in:

    $SYBASE/Installer.log

5   When unloading is complete, log out as the "sybase" user.

    When unloading is complete, eject the CD using the eject command.

    Log in as "root".

6   Unmount the CD-ROM:

171

```
/etc/umount /cdrom
```

Installing components in *cmdfile*

To install components by means of a *cmdfile*, you must begin in GUI mode. To begin the installation process:

1   Verify that you are logged in as the "sybase" user with "root" privileges and that your environment is set up as described in the "Pre-installation tasks" on page 42.

2   Insert the Server CD in the CD-ROM drive.

Log out, log on as "root", mount the CD, log out as "root", and log back on as "sybase".

3   At the UNIX prompt, start the Studio Installer:

```
cd /cdrom
 ./install
```

4   Select the type of installation to be performed. Backup Server, Monitor Server, and XP Server are installed, by default, with Adaptive Server.

   •   *Standard Install* – a standard install installs all the components necessary for most users.

   •   *Full Install* – a full install installs every component on the CD.

   •   *Customized Install* – a customized install allows you to select which components to install. Certain components are automatically installed if they are required to run other selected components.

5   Select a new installation directory, or accept the default.

   Note If you select Customized Install, the next window is the Component Selection screen, which allows you to specify the components to install.

Components that would be installed in a standard installation appear with a check in the check box to the left of the product name. You may select or deselect components from this list. Components with subcomponents have a More... button enabled. Click this button to select or deselect subcomponents.

6   Click Save.

7   Specify a directory. The component information is automatically generated when saving to the *cmdfile*. You may rename the *cmdfile*.

8   After the information is written to the *cmdfile*, the installer returns you to the GUI. Click Cancel, then Yes, to exit the Studio Installer.

9   To complete the installation using the *cmdfile*, go to a command-line prompt at the CD-ROM drive and enter:

```
./install -f path/filename
```

where *path* is the full path to the *cmdfile*, and *filename* is the name of the *cmdfile*.

10 When installation is complete, log out as the "sybase" user.

When installation is complete, eject the CD using the eject command.

Log in as "root".

11 Unmount the CD-ROM:

```
/etc/umount /cdrom
```

12 After Studio Installer is exited, the environment variables are set manually or with the *.sh* and *.csh* script files, automatically downloaded by the Studio Installer. The procedures for using the scripts files are provided in Chapter 4, "Installing Sybase Servers."

13 Check the installation log file to see a record of the installation process. The server log file is located in:

$SYBASE/Installer.log

The PC-client installation log file is in:

%SYBASE%\Installer.log

173

Index

X

XML classes
   and Java classes in   30
XP Server
   described   4
   more information about   62
   starts automatically with first ESP request   93
   unloading   48, 172
xp_cmdshell command   95