SYBASE®

Installation Guide
Sybase® Adaptive Server™ Enterprise

Sun Solaris

Version 12.0

Sybase® Adaptive Server® Enterprise

Document ID: 35889-01-1200-01

Last revised: November 1999

Sybase, Inc., 6475 Christie Avenue, Emeryville, CA 94608.

Contents

About This Book ..... **ix**

CHAPTER 1 **Overview** **1**
- Installation Definitions 1
- User Roles 2
- Product Descriptions 2
- Installation and Setup Utilities 5
- Environment Variables 5
- Sybase Installation Directory 7
- Client/Server Communication (the Interfaces File) 9
- Adaptive Server Devices and System Databases 11
  - master Device 11
  - sybsystemdb Device and Database 11
  - sysprocsdev Device 12
  - Optional Devices and Databases 12
- What's Next? 14

CHAPTER 2 **Installation Requirements** **15**
- System Requirements 15
- Product Disk Space Requirements 17
- Language Module Sizes 18
- Adaptive Server Specifications 19
- What's Next? 22

CHAPTER 3 **Unloading Server Products from Distribution Media** **23**
- Pre-Unloading Tasks 23
  - If You Are Unloading for an Upgrade 25
- Studio Installer 26
  - Unloading Components with Studio Installer: GUI Mode 26
  - Unloading with Studio Installer: NOGUI Mode 30
  - Unloading Components in cmdfile 33
- What's Next? 34

CHAPTER 4 Sybase Software Asset Management (SySAM) 37
What Is SySAM? 37
Adaptive Server Features Licensed Through SySAM 37
How Does SySAM Work? 38
Types of SySAM Systems 40
Setting Up Adaptive Server with SySAM 41
Installing SySAM Software in Your Enterprise 42
Starting the SySAM 43
Enabling SySAM Features 45
Adding Adaptive Server Features 45
Adding Additional Licenses 46
Adding a License to a Pre-Existing Server 47
Starting Adaptive Server with SySAM features 48
Troubleshooting SySAM 49
Authorization Code Input Error 50
What's Next? 52

CHAPTER 5 Installing Sybase Servers 53
Overview 53
Pre-Installation Tasks 54
Determining the Location, Type, and Size of a Database Device 56
Installing Server Components 58
Installation Methods 58
Setting Environment Variables 59
Installation 59
Installation – Adaptive Server 61
Installation – Backup Server 64
Installation – Monitor Server 67
Installation – FTS 70
Viewing the Installation Status Output Screen 71
Changing the Default Language and Character Set 72
Installation Complete 72
Resource File Installation 73
Editing a Resource File 73
Using srvbuildres 77
What's Next? 77

CHAPTER 6 Post-Installation Tasks 79
Post-Installation Tasks 79
Setting the Sybase System Administrator Password 79
Resetting Variables 80
Verifying That Servers Are Running 80

Information Required for XP Server 81
Setting jConnect Environment Variables..... 81
Setting JDBC_HOME 81
Setting CLASSPATH..... 82
Restrictions on Setting CLASSPATH When Using Netscape . 83
Installing Stored Procedures for jConnect..... 83
Adaptive Server Enterprise 84
SQL Anywhere or Adaptive Server Anywhere 84
jConnect Version 4.2 and 5.2..... 84
What's Next?..... 89

CHAPTER 7 Unloading Sybase PC-Client Products from Distribution Media 91
Overview 91
Product Descriptions..... 91
Unloading PC-Client Products 94
Installing the Adaptive Server Plug-In for Sybase Central 96
Installing the Sybase Central Plug-In 97
Installing the Required Microsoft DLL on Windows 95..... 97
Installing a Java Runtime Environment..... 98
Installing the Entire Java Development Kit..... 98
Enabling the Monitoring Functions..... 98
Testing the Sybase Central Installation..... 100
Installing Open Client/C 101
Installing Open Client Runtime for Windows 95/98 and NT .. 101
Installing InfoMaker..... 103
Configuring Network Connections for Client Products 104
Adding a Server to the sql.ini File..... 104
Enabling TCP Connections..... 106
Using Open Database Connectivity 107
What's Next?..... 107

CHAPTER 8 Customizing Localization for Adaptive Server..... 109
Overview of Localization Support..... 109
Language Modules..... 110
Character Sets 111
Default Character Sets for Servers 111
Supported Character Sets..... 112
Character-Set Conversion..... 116
Conversions Between Server and Client..... 117
Limitations of Hankaku Katakana Conversion..... 117
Sort Orders..... 120
Available Sort Orders 120
Gathering Information About Collation 122

Language Modules 126
Installing a New Language Module 126
Message Languages 127
Localization 127
Localization Directories 127
About the Directory 128
About the charsets Directory 128
About the locales.dat File 129
Changing the Localization Configuration 131
For Adaptive Server 131
For Backup Server 132
Sort Orders 133
Character sets 136
charset Utility 137

CHAPTER 9 Upgrading Sybase Servers 139
Overview 139
Pre-Upgrade Tasks 140
Check System and Upgrade Requirements 141
Check RUN_server File Location 142
Test Current Applications and Stored Procedures 142
Check for Reserved Word Conflicts 142
Verify That Users Are Logged Off 144
Check Database integrity 145
Back up Databases 145
Dump Transaction Logs 145
Database and Devices 146
Turn Off Database Options 151
Disable Disk Mirroring 151
Checking Memory Caches 151
Upgrading a Database That Uses Auditing 152
Preparing to Upgrade with Replicated Databases 153
Upgrading to Adaptive Server 12.0 153
Using sqlupgrade 153
Using sqlupgraderes 155
Post-Upgrade Tasks 157
Verifying That Servers Are Running 158
Restoring Functionality in Adaptive Server 158
Reenabling Replication Server 160
Reenabling Auditing 161
Upgrading Backup Server, Monitor Server, and XP Server 162
What's Next? 163

CHAPTER 10 Adding Optional Functionality to Adaptive Server 165
Installing Auditing 165
Audit System Devices and Databases 166
Overview of Audit Installation 167
Pre-Installation Tasks 167
Audit Installation 168
Installing Sample Databases 174
Default Devices for Sample Databases 174
interpubs Database 176
jpubs Database 177
Maintaining the Sample Databases 178
What's Next? 178

CHAPTER 11 Starting and Stopping Servers 179
Overview 179
Starting Servers 180
Using the Runserver File 180
Using the startserver Command 180
Using the monserver Command 181
Starting Servers When the Operating System Restarts 181
Stopping Servers 183
Stopping Adaptive Server 183
Stopping Backup Server 184
Stopping Monitor Server 184
Using the kill Command 185
Shutdown and Shared Memory Files 186

CHAPTER 12 Troubleshooting 187
Overview 187
Error Log Locations 188
For Installation Utilities 188
For Sybase Servers 189
Solutions to Common Installation Problems 189
You Do Not Have a 17-Inch Monitor 189
Fatal Error: Cannot Map libct.so 190
Cannot Eject the CD from the Drive 190
DISPLAY Environment Variable Not Set Correctly 190
Client Not Authorized to Connect to Server 191
Address Already in Use 191
Adaptive Server Failed to Boot 192
Cannot Start XP Server 192
Configuration Failed 192
Stopping Adaptive Server After a Failure 192

Recovering from a Failed Installation 193
Removing an Existing Adaptive Server 193
Recovering from a Failed Upgrade 194

CHAPTER 13 Configuring the Operating System 197
Setting Environment Variables 197
DSLISTEN 198
DSQUERY 198
PATH 199
Using the stty Setting 199
Restoring Correct Permissions 199
File Descriptors and User Connections 199
For Sun Solaris 201
Adjusting the Client Connection Timeout Period 203
For Sun Solaris 203
Checking for Hardware Errors 203
For Sun Solaris 204
Monitoring the Use of Operating System Resources 204
For Sun Solaris 204
A Sample C Shell Maintenance Script 204

APPENDIX A **Preparing to Upgrade Servers with Replicated Databases 207**
Suspending Transaction Processing and Replication Activities 208
Draining the Transaction Logs for Primary Databases 209
Draining the RSSD Transaction Log 209
Disabling the Log Truncation Point 210
Draining Each Replicated Primary 211
Restoring Replication After Upgrade 212
Upgrading Adaptive Server 213

INDEX 215

About This Book

This guide, *Installation Guide Sybase Adaptive Server Enterprise on Son Solaris* 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 installers 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, "Unloading Server Products from Distribution Media" describes how to unload Adaptive Server products from the distribution media to your system.
- Chapter 4, "Sybase Software Asset Management (SySAM)" provides installation instructions for Sybase Software Manager (SySAM).
- Chapter 5, "Installing Sybase Servers" describes pre-installation procedures, and how to install server components.
- Chapter 6, "Post-Installation Tasks" describes how to set environment variables and initialize features.

- Chapter 7, “Unloading Sybase PC-Client Products from Distribution Media” describes how to install Adaptive Server client products like the Adaptive Server plug-in to Sybase Central.
- Chapter 8, “Customizing Localization for Adaptive Server” provides instructions for changing the language used for server messages.
- Chapter 9, “Upgrading Sybase Servers” describes how to upgrade an existing Adaptive Server to the current release.
- Chapter 10, “Adding Optional Functionality to Adaptive Server” describes how to install additional (optional) Adaptive Server functionality such as auditing, sample databases that can be used for learning Transact-SQL®, and two-phase commit.
- Chapter 11, “Starting and Stopping Servers” describes the different methods of starting and stopping Adaptive Server, Backup Server™, and other servers.
- Chapter 12, “Troubleshooting” provides installation error messages and possible solutions to installation problems.
- Chapter 13, “Configuring the Operating System” discusses the operating system configuration settings that you can adjust after installing or upgrading Adaptive Server.
- Appendix A, “Preparing to Upgrade Servers with Replicated Databases” provides instructions for preparing to upgrade a pre-11.0 Adaptive Server that has replicated databases.

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 Adaptive Server installation documentation for your platform – describes installation, upgrade, and configuration procedures for all Adaptive Server and related Sybase products.
- *What’s New in Adaptive Server Enterprise?* – describes the new features in Adaptive Server version 12, 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.
- *Adaptive Server 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 Programs* manual for your platform – documents the Adaptive Server utility programs, such as **isql** and **bcp**, which are executed at the operating system level.
- *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 and user-defined functions 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.

- *XA Interface Integration Guide for CICS, Encina, and TUXEDO* – provides instructions for using Sybase's DTM XA Interface with X/Open XA transaction managers.
- *Adaptive Server Glossary* – defines technical terms used in the Adaptive Server documentation.
- *Sybase jConnect for JDBC Programmer's Reference* – describes the jConnect for JDBC product and explains how to use it to access data stored in relational database management systems.

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 technical documents and is included with your software. The DynaText browser (included on the Technical Library CD) 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 Technical Library.
- 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'll 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://sybooks.sybase.com.

Sybase certifications on the web

Technical documentation at the Sybase web site is updated frequently.

❖ **For the latest information on product certifications and/or the EBF Rollups:**

1. Point your web browser to Technical Documents at http://techinfo.sybase.com.
2. In the Browse section, click on What's Hot.
3. Select links to Certification Reports and EBF Rollups, as well as links to Technical Newsletters, online manuals, and so on.

❖ **If you are a registered SupportPlus user:**

1. Point your web browser to Technical Documents at http://techinfo.sybase.com.

2 In the Browse section, click on What's Hot.

3 Click on EBF Rollups.

You can research EBFs using Technical Documents, and you can download EBFs using Electronic Software Distribution (ESD).

4 Follow the instructions associated with the SupportPlus^{SM} Online Services entries.

❖ **If you are not a registered SupportPlus user, and you want to become one:**

You can register by following the instructions on the Web.

To use SupportPlus, you need:

1 A Web browser that supports the Secure Sockets Layer (SSL), such as Netscape Navigator 1.2 or later

2 An active support license

3 A named technical support contact

4 Your user ID and password

❖ **Whether or not you are a registered SupportPlus user:**

You may use Sybase's Technical Documents. Certification Reports are among the features documented at this site.

1 Point your web browser to Technical Documents at http://techinfo.sybase.com

2 In the Browse section, click on What's Hot.

3 Click on the topic that interests you.

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.

CHAPTER 1

Overview

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

Topics covered include:

Name	Page
Installation Definitions	1
User Roles	2
Product Descriptions	2
Installation and Setup Utilities	5
Environment Variables	5
Sybase Installation Directory	7
Client/Server Communication (the Interfaces File)	9
Adaptive Server Devices and System Databases	11
What's Next?	14

Installation Definitions

In this book, these terms are defined as follows:

- *Unload* – to copy Adaptive Server products from the distribution media to your machine, using the Studio Installer.
- *Install* – to make the Adaptive Server products fully functional by configuring various attributes.
- *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.

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.
- *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 Adaptive Server package. For a description of the products on your PC-Client CD, see Figure 7-1 in Chapter 7.

Table 1-1: Product descriptions

Product	Description
Adaptive Server	The relational database server. The default unloading process includes: • Adaptive Server • Backup Server • Full Text Search Server • Monitor Server • XP Server • Adaptive Server utilities • Scripts and configuration files

Product	Description
Adaptive Server Enterprise Monitor	An Open Server™ application that obtains performance statistics on Adaptive Server and makes those statistics available to Monitor Server client applications. Adaptive Server Enterprise Monitor (also called Monitor Server) includes: • Monitor Server for Adaptive Server Enterprise 12.0 – an Open Server application that obtains performance statistics on Adaptive Server and makes those statistics available to monitors in Sybase Central, Monitor Historical Servers, and applications built with Monitor Client-Library. Monitor Server for SQL Server 11.0.x is also available. • Adaptive Server plug-in for Sybase Central – this graphical user interface obtains Adaptive Server performance data from Monitor Server and displays the data, in real time, in tables and graphs. • 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.
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 up to 32 dump devices (this is called *dump striping*) in parallel to dump or load a single database or transaction log. • 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 unloaded, by default, when you unload Adaptive Server software from the distribution media. Use the srvbuild utility to set up Backup Server and to connect Backup Server and Adaptive Server through the interfaces file.

Product	Description
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 Chapter 8, "Customizing Localization for Adaptive Server".
Language Modules (Connectivity)	Provides messages and support files for running Open Client™ applications in various languages.
jConnect™ for JDBC 4.2	Provides a Java database connectivity (JDBC) driver that works with both Sun and Microsoft virtual machines (VM). For more information jConnect for JDBC, see the jConnect product page at http://www.sybase.com/products/internet/jconnect/.
jConnect for JDBC 5.2	Provides a Java database connectivity (JDBC) driver that works with both Sun and Microsoft virtual machines (VM). For more information jConnect for JDBC, see the jConnect product page at http://www.sybase.com/products/internet/jconnect/.
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 Chapter 8, "Customizing Localization for Adaptive Server".	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 *Utility Programs for UNIX Platforms*.	Noninteractive, file-based interface
sqlupgrade	Upgrade Adaptive Server.	X-Windows/Motif GUI
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. • Unload Adaptive Server products from distribution media.	X-Windows/Motif GUI

Environment Variables

It is crucial to the operation of Sybase products that the system environment variables be set correctly.

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). The installation instructions in this guide explain when to set these variables.

- DSLISTEN – Defines the name Adaptive Server uses to listen for client connections if no name is given during the Adaptive Server start-up. If DSLISTEN is not set, and no name is given during the Adaptive Server start-up, the Adaptive Server name defaults to the server name given at installation.
- DSQUERY – Defines the Adaptive Server name that client programs try to connect to if no Adaptive Server is specified with a command-line option. If DSQUERY is not set, and you do not supply the Adaptive Server name with a command-line option, clients attempt to connect to the server name given at installation.
- 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 subdirectory directory of the Adaptive Server components.
- SYBASE_OCS – Defines the subdirectory to which the Open Client is set.
- SYBASE_ FTS – Defines the subdirectory for the full-text search components.
- SYBASE_SYSAM – Points to the license-management software directory.
- LM_LICENSE_FILE – Points to the *license.dat* file in the *SYBASE_SYSAM/licenses*/ directory.
- PATH – Specifies which directory path to search for executables.The Sybase executables are located in the installed components */bin* directory.

  When using the source files: S*YBASE.csh* or *SYBASE.sh*, the following paths are prepended to the PATH environment variable:

  $SYBASE/$SYBASE_ASE/bin:$SYBASE/$SYBASE_OCS/bin
- LD_LIBRARY_PATH – Specifies which directory to search for executables. The Sybase executables are located in the installed component's */bin* directory.

  When using the source files: SYBASE.csh or SYBASE.sh, the following paths are prepended to the LD_LIBRARY_PATH environment variable:

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

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 1-3 illustrates part of the Adaptive Server installation directory structure.

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

Table 1-3: Installation directory for $SYBASE

Component Directory	Subdirectory	Subdirectory
/ASE-12_0		
	/bin	
	/debugger	
	/init	/auditinit
		/bsrv
		/logs
		/sqlsrv
	/install	/spr
	/sample	/esp
		/JavaSql
		/server
	/scripts	
	/upgrade	
/charsets		
/collate	/unicode	
/config		
/FTS-12_0		
/installed		
/Installer	/bin	
	/lib	
/jConnect_docs	/docs	
/jConnect-4_2	/classes	/com /gateway /sample
	/devclasses	/com
	/docs	/en
	/gateway	
	/sample	
	/sp	
	/tools	
	/devclasses	
/jConnect-5_2	/classes	/gateway2 /sample2
	/devclasses	
	/docs	/en
	/gateway2	

Component Directory	Subdirectory	Subdirectory
	/sample2	
	/sp	
	/tools	
/jre-1_1-NT386	/bin	
	/lib	/security
/jutils-2_0	/cascade	/uk
	/jisql	/doc /HelpFiles
	/ribo	/doc
/locales	/english	
	/message	
	/unicode	
/OCS-12_0	/bin	
	/config	
	/include	
	/lib	
	/locales	
	/sample	
	/scripts	
/SQLRemote	/bin	
	/\scripts	
/SYSAM-1_0	/bin	
	/licenses	

Client/Server Communication (the Interfaces File)

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.

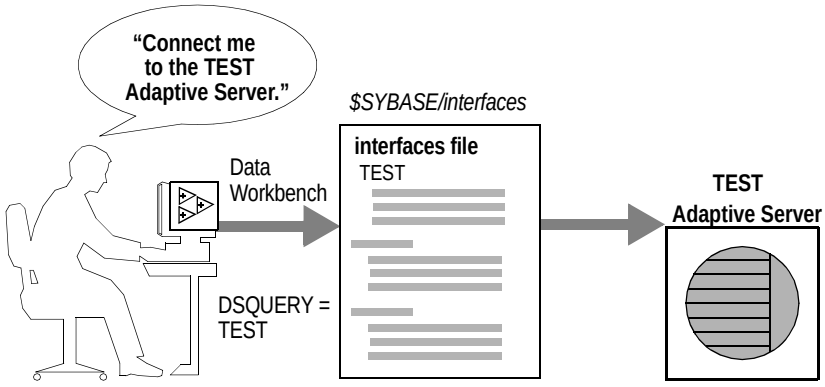
For Sybase products to interact with one another, each product needs to know where the others reside on the network. This information is stored in an interfaces file, named *interfaces* on UNIX platforms, located in the $SYBASE installation directory. After your Adaptive Server or client software is installed, it can connect with any server on the network that is listed in the interfaces file.

The interfaces file is like an address book. It lists the name and address of every known server. When you are using a client program, and you want to connect with a particular server, the client program looks up the server name in the interfaces file and connects to that server, as shown in Figure 1-1. You can supply the name of the server by using the DSQUERY environment variable.

On TCP/IP networks, the port number gives clients a way to identify the Adaptive Server to which they want to connect. It also tells Adaptive Server where to listen for incoming connection attempts from clients. Adaptive Server uses a single port for these two services (referred to as *query service* and *listener service*).

On SPX networks, the socket number gives clients and servers a way to identify each other.

Figure 1-1: Communicating with a server



During installation, you use the **srvbuild** utility to create a new server. The **srvbuild** process adds entries to the *interfaces* file for your new Adaptive Server, Backup Server, Monitor Server, and XP Server.

For instructions on using **dsedit** or **dscp** to modify interfaces file entries or to create new interfaces file entries for existing servers, see Chapter 13, "Configuring the Operating System".

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 13.

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 it 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.

See Chapter 9, “Upgrading Sybase Servers” for instructions on how to create the *sybsystemdb* device and database for two-phase commit and Data Transfer Management.

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

The sample databases are stored on the master device at installation, but should be moved to a user-defined device after installation. For information on the sample databases, see “Optional Devices and Databases” on page 12.

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 Chapter 10, "Adding Optional Functionality to Adaptive Server" 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 Chapter 10, "Adding Optional Functionality to Adaptive Server".

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

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.

Information on installing *dbccdb* is located in the chapter "Checking Database Consistency," in the *System Administration Guide*.

What's Next?

Do any of the following:

- For Adaptive Server product descriptions, and server and database specifications, see Chapter 2, "Installation Requirements".
- For information on unloading software from the distribution media, see Chapter 3, "Unloading Server Products from Distribution Media".
- For information on Sybase Software Asset Management, see Chapter 4, "Sybase Software Asset Management (SySAM)".
- To install Adaptive Server 12.0, see Chapter 5, "Installing Sybase Servers".
- To upgrade from a previous version to Adaptive Server 12.0, see Chapter 9, "Upgrading Sybase Servers"

CHAPTER 2

Installation Requirements

This chapter covers:

Name	Page
System Requirements	15
Product Disk Space Requirements	17
Language Module Sizes	18
Adaptive Server Specifications	19
What's Next?	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 release-level, RAM, and network-protocol requirements for Adaptive Server. See the release bulletin for the latest information on required system patches.

Table 2-1: System requirements for Adaptive Server

Platform and Operating System	Release Level	RAM Required	Network Protocol
Sun Solaris (SPARC) (Certified on Sun Solaris 2.6)	2.6	32MB	TCP or SPX

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

Table 2-2: System requirements for jConnect on UNIX

Hardware	Any UNIX platform that supports Java VM 1.1.x
Operating system	Sun Solaris, HP UX, IBM AIX, DEC Digital Unix, SGI IRIX, etc.
Java Developer's Kit	Sun's JDK 1.1.x or later
Database	• Adaptive Server Enterprise, from SQL Server System 10, version 10.0.2, 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 4.2 and 5.2 require Adaptive Server Enterprise 12.x or later: • 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, Windows 98, and Windows 95. Table 2-3 lists the RAM required for client products.

Table 2-3: RAM requirements for client products

Product	RAM Required
Adaptive Server plug-in for Sybase Central	12MB (Windows 95) 16MB (Windows NT)
Monitor Client-Library	16MB (32MB recommended)
Open Client/C	16MB
jConnect for JDBC 4.2	16MB
jConnect for JDBC 5.2	16MB
jutils (Cascade, jisql, Ribo)	N/A

Product Disk Space Requirements

Table 2-4 lists the disk space requirements for each product.

Table 2-4: Product disk space requirements

Product and Version Level	Disk Space Required
Adaptive Server 12.0	140MB
jConnect 4.2	4MB
jConnect 5.2	4MB
jutilities (Cascade Gateway, jisql, Ribo)	1.5MB
jConnect Documentation	.5MB
Open Client/C 12.0 SDK	52MB
Language Modules for Adaptive Server 12.0	2–7MB per language
Language Modules (Open Client 12.0)	3–6MB per language
Adaptive Server Monitor Server 12.0	1.6MB
Monitor Server 11.5.1 for SQL Server 11.0.x	1.1MB
Monitor Historical Server 12.0	2.9MB
Monitor Client Library 12.0	1.8MB

Table 2-5 shows the installation requirements for the Adaptive Server plug-in for Sybase Central.

Table 2-5: Installation requirements for plug-in

Operating System	Disk Space
Windows NT 4.0 or Windows 95/98 Service Pack 3 or later	12MB

Language Module Sizes

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

Table 2-6: Language module sizes

Language	Size of Adaptive Server Language Module	Size of Open Client Language Module
Brazilian Portuguese	6.5MB	1.2MB
Chinese (simplified)	2MB	1.2MB
Chinese (traditional)	2MB	1.2MB
French	6.5MB	1.5MB
German	6.5MB	1.5MB
Japanese	4.5MB	1.2MB
Korean	2MB	1.2MB
Spanish	6.5MB	1.2MB

Adaptive Server Specifications

This section provides Adaptive Server specifications for Sun Solaris.

Table 2-7: Adaptive Server specifications for Sun Solaris

Hardware		
Processor	(32-bit) [64-bit]	
Maximum CPUs per server	128	
Minimum RAM for Adaptive Server	(64MB) [92MB]	(32,768 2K pages) [47,104 2K pages]
Maximum login IDs per server	64K	
Minimum RAM per additional user	(111K) [173K]	With default stack size, packet size, and user log cache size
Default user stack size	(46,080 bytes) [86,016 bytes]	Minimum 26,624 bytes
Database Specifications		
Databases per Adaptive Server	32,767	Practical limit is approximately 100
Default database size	2MB	Minimum 2MB
Maximum database size	2^{42} (4TB)	231 pages *2K page size

Maximum size of a database device (disk partition)	2^{35} (32GB)	• Requires Solaris 2.6 or later • 2GB maximum device size if using operating system files • 32GB device size with operating system files will be supported with Solaris 2.7
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	
Number of users per database	48K	
master database	6MB	Minimum required for a new installation
	7MB	Minimum required for an upgrade
model database	2MB	Minimum required for a new installation
	3MB	Minimum required for an upgrade
sybsystemdb database	4MB	Minimum required for an upgrade
	20MB	Recommended. Approximately 25% of the database should be data storage, and 75% should be log storage.
tempdb database	2MB	Minimum required for a new installation
	3MB	Minimum required for an upgrade
sybsystemprocs database	80MB	Minimum required for an upgrade
pubs2 database	3MB	Minimum required for a new installation
	4MB	Minimum required for an upgrade
pubs3 database	3MB	Minimum required for a new installation
	4MB	Minimum required for an upgrade
interpubs database	2MB	Minimum required for a new installation
	3MB	Minimum required for an upgrade
jpubs database	2MB	Minimum required for a new installation
	3MB	Minimum required for an upgrade

Table Specifications		
User objects per database	$2^{31} - 100$	
Columns per table	250	
Indexes per table	250 (one clustered index)	
Rows per table	Limited by available storage	Maximum 2^{32}
Maximum user data per row	1960 bytes for all pages locked table	1958 bytes for data-only-locked tables with fixed-length columns
Maximum number of data bytes per text/image page	1800 bytes	
Columns per composite index	31	
Bytes per index key	600	
Creation of clustered index	Free space needed on segment is 1.2 * (size of table)	For sorted data, approximately 20% 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 it not counted) correlations and self-joins. Maximum of 14 work tables.
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) [4K]
Maximum number of parameters per stored procedure	255

What's Next?

To unload Adaptive Server software to your system, see Chapter 3, "Unloading Server Products from Distribution Media".

CHAPTER 3

Unloading Server Products from Distribution Media

This chapter describes how to unload Adaptive Server Enterprise server products from the Sybase distribution media onto your system.

Topics covered include:

Name	Page
Pre-Unloading Tasks	23
Studio Installer	26
Unloading Components in cmdfile	33
What's Next?	34

Pre-Unloading Tasks

This section describes tasks that you need to complete for:

- Adaptive Server Enterprise products
- jConnect 4.2, 5.2, and Java utilities

Adaptive Server Enterprise products

1 Read the release bulletins for the latest information on the products (Adaptive Server, Monitor Server, and so on) that you are unloading.

  Your Sybase product shipment includes printed release bulletins; they are also available

  at http://www.sybase.com

2 Create a "sybase" account on your system to perform all unloading tasks. The operating system administrator usually sets up this account, as this requires "root" privileges.

3 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 unload, installation, upgrade, and setup 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.

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

If you are unloading for an upgrade, see "If You Are Unloading for an Upgrade" on page 25.

5 Verify that the directory location for the Sybase installation has sufficient space to accommodate unloading the software.

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

Note As part of the installation, the Studio Installer sets most of the environments 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 5, "Installing Sybase Servers".

Also note that the Studio Installer does not set jConnect or Java utility environment variables—you must set them manually using the instructions in Chapter 6, "Post-Installation Tasks"

jConnect 4.2, 5.2, and Java utilities

Before you install jConnect version 4.2, 5.2, jisql, or Ribo:

1 Install the Java Development Kit (JDK). You can download the current JDK from Sun's Java Software Web site:

at http://java.sun.com/products/index.html

then install it using the *JDK Installation Instructions*.

You can also run the installer using jView, which you can download from:

at http://www.microsoft.com/java/download.htm

When you install jConnect 5.2, you receive JDBC 2.0 compatible classes, regardless of what 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	jConnect 4.x

If you are developing...	Use JDK Version...	Use jConnect Version
JDBC 2.x applications/servlets	1.2.x or compatible browser	jConnect 5.x

2 Set the JAVA_HOME environment variable.

After installing the JDK, define the JAVA_HOME environment variable to point to the JDK installation directory. For example, if you installed JDK 1.2 (Java 2) the setting might look like:

```
$JAVA_HOME/jdk1.2 (on UNIX)
```

See your operating system documentation for further instructions.

If You Are Unloading for an Upgrade

To unload components for the purpose of upgrading a product:

1 Back up all existing Sybase installations. Unloading into an existing Sybase installation directory overwrites the previous installation.

2 Unload the new software into a different directory from your currently installed version. Verify the new location in the Sybase Release Directory box in the Studio Installer main menu.

Warning! If you unload Adaptive Server 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.

3 If you are unloading Monitor Server for SQL Server 11.0.x, verify that you have write permissions on the current Monitor Server files. The files are:

- *$SYBASE/$SYBASE_ASE/bin/monserver_110x*
- *$SYBASE/$SYBASE_ASE/scripts/installmon*
- *$SYBASE/$SYBASE_ASE/install/setperm_monserv*

4 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. Table 3-1 shows which upgrade paths require you to change and recompile the source code.

Table 3-1: Upgrade paths that require recompiling

Upgrading From jConnect Version	To jConnect Version				
	4.0	*4.1*	*4.2*	*5.0*	*5.2*
3.0	No changes required	No changes required	No changes required	Changes Required	Changes Required
4.0	N/A	No changes required	No changes required	Changes Required	Changes Required
4.1	N/A	N/A	No changes required	Changes Required	Changes Required
4.2	N/A	N/A	N/A	Changes Required	Changes Required
5.0	N/A	N/A	N/A	N/A	No changes required

Studio Installer

The Studio Installer is Java-based and uses XML input. It loads and installs all Sybase components across all platforms.

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

There are three options for unloading Sybase components from the distribution media via the Studio Installer: the default GUI (graphical user interface) mode, a NOGUI mode, or by a *cmdfile*.

After the unload is complete, you can install and configure the components by following the instructions in Chapter 5, “Installing Sybase Servers”.

Unloading Components with Studio Installer: GUI Mode

This section describes how to install Adaptive Server components on the Primary Network Node. Information on installing Adaptive Server and PC-client components in the networked enterprise are discussed in Chapter 4, “Sybase Software Asset Management (SySAM)”.

The installation procedure assumes that:

- You have completed the list of tasks in “Pre-Unloading Tasks” on page 23.

- The target computer meets the requirements outlined in Chapter 2, “Installation Requirements”.

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

Note 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 5, “Installing Sybase Servers”.

Studio Installer does not set jConnect or Java utility environment variables—you must set them manually using the instructions in Chapter 6, “Post-Installation Tasks”.

During the installation process, the computer must be rebooted. After rebooting, the computer accesses the CD for additional data. For this reason, the CD must remain in the CD-ROM drive during reboot, and the CD-ROM drive should be on the installation machine.

To unload Server components:

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-Unloading Tasks” section of this chapter.

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

3 The Solaris operating system automatically mounts the CD.

If you get CD-reading errors, check your operating system kernel to make sure the ISO 9660 option is turned on.

4 At the UNIX prompt, start the Studio Installer:

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

where *cdrom* is the directory you specified when mounting the CD-ROM drive, and *./install* unloads the components in the GUI mode.

5 Select the type of installation to be performed. Backup Server, Monitor Server, and XP Server are unloaded, by default, with Adaptive Server.

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

- *Full Install* – installs every component on the CD.
- *Customized Install* – allows you to select which components to install. Certain components will automatically be installed if they are required to run other selected components.

6 Enter the target directory. If the target directory does not already exist, the Studio Installer prompts you to create it. Click Yes to proceed.

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. Clicking this button allows you to select or deselect subcomponents.

---

7 Click Next to launch the Summary screen.

The Summary screen displays every component that will be installed by the Studio Installer. Disk space required for each selected component is also shown, along with the available disk space.

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

---

8 Click Next to continue the installation. The Studio Installer unloads the components from the CD and displays a progress indicator.

If no component has been selected, the Studio Installer generates an error message and stops the installation. Click Back to select components, or Cancel to cancel the installation procedure.

Note Optionally, you may select Save from the Summary screen to save all the installation information into a *cmdfile* to proceed with the installation in a non-interactive, silent install. (See "Unloading Components in cmdfile" on page 33.)

If any components selected to be installed require a license, the Sybase Software Asset Management (SySAM) screen displays and prompts you for certification information at this time. If you install any components without the appropriate license information, only Adaptive Server 12.0 without licensed features is enabled.

SySAM License Manager

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

   - Click Yes to enter license information from the Sybase Software Asset Management Certificate. Proceed to step 2.
   - Click No if you do not have licenses for optional Adaptive Server features.The installer prompts: "Have you registered Sybase Software Asset Management Certificates at a central license host?"

     Note You are not prompted for further licensing information, and you will not be able to use any optional features in Adaptive Server if you select No at this point.

   - Click No. See "Installing SySAM Software in Your Enterprise" on page 42 for information about configuring client machines to access Adaptive Server licenses from the primary network node.

     The Studio Installer leads you through the installation of Adaptive Server. Proceed to Chapter 5, "Installing Sybase Servers"

2 Enter information from the Sybase License Certificate for each Adaptive Server feature you have purchased:

   - **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_HA**, **ASE_DTM**, and **ASE_ASM**.

     See Table 4-1 on page 38 for a description of features.

- **Feature Count:** Enter your license count number.
- **Software Version:** Enter the Adaptive Server software version.
- **Authorization Code:** Enter the license key for the purchased feature.

Note Entries are case sensitive. Enter the values exactly as they appear on the Sybase Software Asset Management Certificate. See Chapter 4, “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.

Note 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*.

Unloading with Studio Installer: NOGUI Mode

To unload components in a NOGUI environment, perform all pre-installation preparation steps as described; however, replace the window-based Studio Installer instructions (“Unloading Components with Studio Installer: GUI Mode” on page 26) 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 filename 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 At the prompt, type:

```
cd/cdrom
./install -c
```

2 Select the type of install to perform.

- Full Install
- Standard Install
- Cancel

3 Select one of the following:

- Continue – to continue.
- Previous Menu– to return to the previous screen.
- Cancel – to terminate the installation procedure.

4 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, seeChapter 4, “Sybase Software Asset Management (SySAM)”.

Completing the Unload Process

1 Follow the directions on the screen.

2 When the unload is complete, use the **asecfg** utility to:

- Configure a New Server – launches the **srvbuild** utility (Chapter 5, “Installing Sybase Servers”).
- Upgrade an Existing Server – launches the **sqlupgrade** utility (Chapter 9, “Upgrading Sybase Servers”).
- Localize an Existing Server – launches the **sqlloc** utility (Chapter 8, “Customizing Localization for Adaptive Server”).
- 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 Utilities 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, remove the CD from the drive:

```
/usr/bin/eject cd
```

You have successfully unloaded Adaptive Server onto your system; however, you do not yet have a fully functional product. To continue, see “What’s Next?” on page 34.

Unloading Components in *cmdfile*

To unload components by means of a *cmdfile*, you must begin in GUI mode. To begin the unload 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-Unloading Tasks" on page 23.

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

The Solaris operating system automatically mounts the CD.

If you get CD-reading errors, check your operating system kernel to make sure the ISO 9660 option is turned on.

3 At the UNIX prompt, start the Studio Installer:

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

where *cdrom* the directory you specified when mounting the CD-ROM drive, and *./install* unloads the components in the GUI mode.

4 Select the type of installation to be performed. Backup Server, Monitor Server, and XP Server are unloaded, 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 will automatically be 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 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. Clicking this button allows you 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 and terminate the installation

9 To complete the installation using the *cmdfile*, go 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 *name* is the name of the *cmdfile*.

10 When unloading is complete, remove the CD from the drive:

```
/usr/bin/eject cd
```

11 You have successfully unloaded Adaptive Server onto your system; however, you do not yet have a fully functional product. To continue, see "What's Next?" on page 34.

12 Unmount the CD-ROM:

```
/etc/umount /cdrom
```

13 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 environment files are provided in Chapter 5, "Installing Sybase Servers"

14 Check the installation log file to see a record of the installation process. The server log file is located in:

   $SYBASE/Installer.log

For information on licensing features, see Chapter 4, "Sybase Software Asset Management (SySAM)".

What's Next?

You have successfully unloaded Adaptive Server and its products onto your system. To determine your next step, see the following table:

To Do This	See
Perform a first-time configuration of: • Adaptive Server • Backup Server • Monitor Server • XP Server	Chapter 5, “Installing Sybase Servers”
Install client products	Chapter 7, “Unloading Sybase PC-Client Products from Distribution Media”
Upgrade an existing Adaptive Server	Chapter 9, “Upgrading Sybase Servers”
Configure and use Historical Server	*Sybase Adaptive Server Enterprise Monitor Historical Server User’s Guide*
Configure and use Monitor Client-Library	*Sybase Adaptive Server Enterprise Monitor Client-Library Programmer’s Guide*

CHAPTER 4

Sybase Software Asset Management (SySAM)

This chapter describes how to license optional Adaptive Server features, and add new licenses to existing servers using Sybase Software Asset Management (SySAM).

Topics covered are:

Name	Page
What Is SySAM?	37
Setting Up Adaptive Server with SySAM	41
Starting the SySAM	43
Adding Adaptive Server Features	45
Troubleshooting SySAM	49
What's Next?	52

What Is SySAM?

SySAM is a licensing mechanism that:

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

Adaptive Server 12.0 uses SySAM to license optional features.

Adaptive Server Features Licensed Through SySAM

Table 4-1 describes the Adaptive Server features that are licensed using SySAM.

Table 4-1: 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.
DistributedTransaction Management	ASE_DTM	Distributed transaction management features, including the Adaptive Server version 12.0 XA interface.
Java in Adaptive Server	ASE_JAVA	Java and XML classes in Adaptive Server databases.
Advanced Security Mechanisms	ASE_ASM	Network-based authentication and encryption using DCE and CyberSafe.

Note The basic Adaptive Server Enterprise license must be registered before any of the optional features can be enabled.

How Does SySAM Work?

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 products with SySAM support
- One or more license files
- Asset management software, which consists of a license management daemon and a Sybase daemon

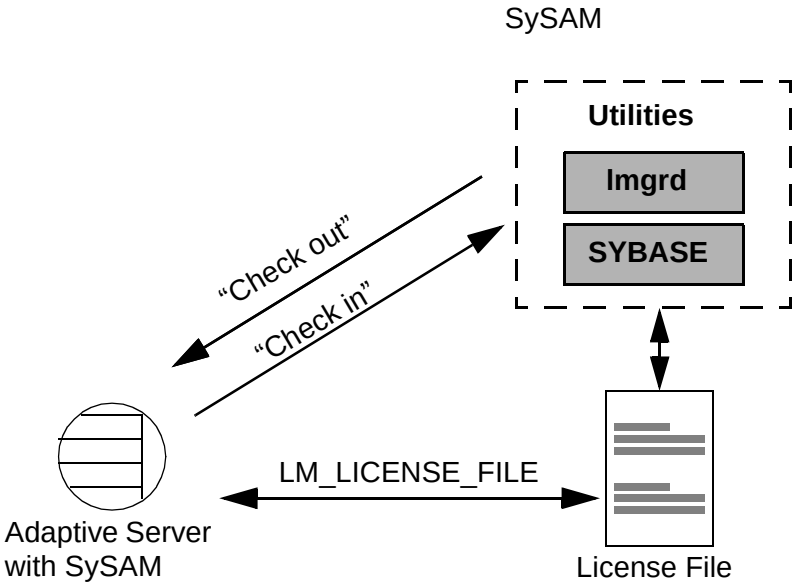
Figure 4-1 shows the relationship between these components.

Adaptive Server Check-Out Procedure

When you boot Adaptive Server 12.0, it attempts to locate a license file named *$SYBASE/$SYBASE_SYSAM/licenses/license.dat*. If the file does not exist, Adaptive Server looks for the license file specified in the environment variable **LM_LICENSE_FILE**.

The license file contains a pointer to the SySAM software, or to a remote machine on which SySAM is running. SySAM consists of two daemons—the license management daemon, **lmgrd**, and the **SYBASE** daemon—and utilities to manage licensing activities, **lmutil**. The daemons handle requests to check in or check out licensed features, as shown in Figure 4-1.

Figure 4-1: SySAM Components



Using information in the license file, Adaptive Server connects to SySAM and attempts to check out a license for the base Adaptive Server product, ASE_SERVER. If the ASE_SERVER license is checked out successfully, Adaptive Server continues to boot 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 or high availability, it attempts to check out licenses for those features during the boot-up process. If a license is not available for an optional feature, Adaptive Server still starts, but the feature cannot be used.

Types of SySAM Systems

The basic components of SySAM can be arranged for a single standalone Adaptive Server, or for a collection of servers on a network.

Standalone System

In standalone systems, SySAM components reside on the same computer that runs Adaptive Server. Standalone SySAM systems can license either Adaptive Server or Adaptive Server with optional features.

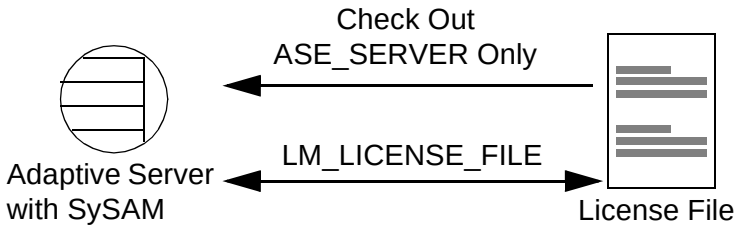
Standalone System with Optional Features

In a standalone system where you have licensed optional Adaptive Server features, *all* of the SySAM components shown in Figure 4-1 reside on a single machine. The license file points directly to the asset management software, which runs on the same machine as Adaptive Server.

Standalone System with No Optional Features

In a standalone installation where no optional Adaptive Server features are used, you do not need to run the asset management software. See Figure 4-2. In this system, the license file contains all license information for ASE_SERVER. Adaptive Server obtains the base license from the local license file and *does not* attempt to contact the license daemon software. No optional features can be used in this configuration.

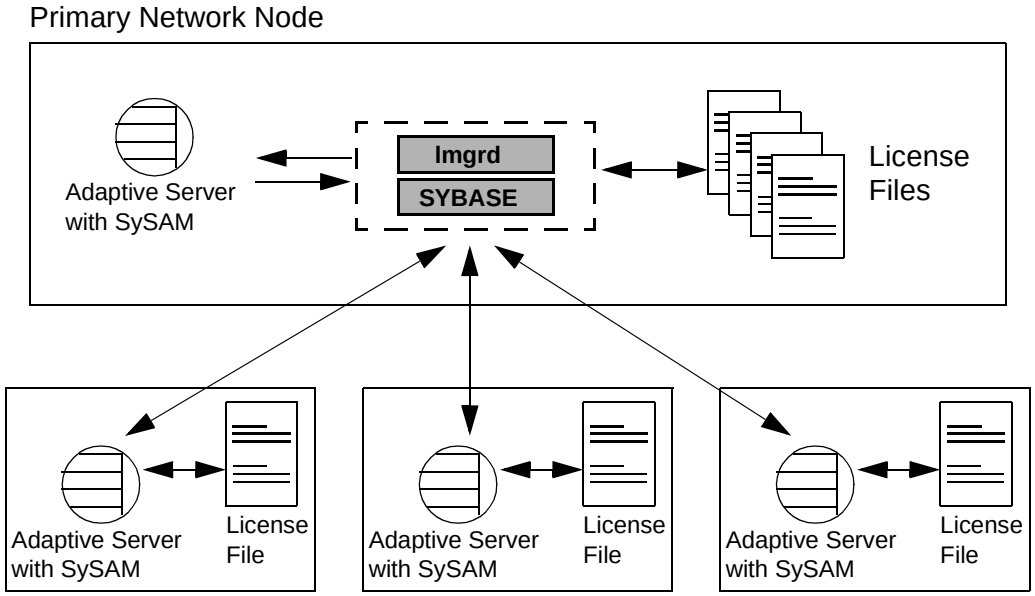
Figure 4-2: Standalone system without optional features



Network System

In a network system, a single machine called the **primary network node** runs SySAM. The software on the primary network node handles check in/check out requests from all other Adaptive Server instances in the system, as shown in Figure 4-3.

Figure 4-3: Network system



In a network system, the license files for remote Adaptive Server installations contain only pointers to the primary network node. Servers check out individual licenses using the license files and asset management software that reside on the primary network node.

Setting Up Adaptive Server with SySAM

The steps required to enable SySAM to track the functionality of Adaptive Server depend on the type of asset management system you want to use and on the Adaptive Server features you have purchased. Table 4-2 guides you through the setup process.

Table 4-2: Steps for setting up Adaptive Server for use with SySAM

If You Want To Use:	And You Have Purchased:	Follow These Instructions:
Standalone Asset Management	• Adaptive Server base product, but • *No* additional licensed features	"Installing SySAM Software in Your Enterprise" on page 42
	• Adaptive Server base product, and • One or more licensed Adaptive Server features	1 "Installing SySAM Software in Your Enterprise" on page 42 2 "Starting the SySAM" on page 43

If You Want To Use:	And You Have Purchased:	Follow These Instructions:
Network Asset Management	• Adaptive Server base product, with or without additional, licensed features	1 "Installing SySAM Software in Your Enterprise" on page 42 2 "Starting the SySAM" on page 43 3 "Installing SySAM Software in Your Enterprise" on page 42

Installing SySAM Software in Your Enterprise

Whether you want to run a networked software asset management system or a standalone system with licensed Adaptive Server features, you must first enable the asset management software (SySAM) that was installed during the installation of Adaptive Server.

SySAM is already enabled if you licensed the Sybase Adaptive Server and other registered components during the unloading as described in Chapter 3, "Unloading Server Products from Distribution Media".

You must manually enable SySAM if:

- You elected not to register licensed products during the installation, or
- You are adding additional component licenses after the initial installation of Adaptive Server.

Use the instructions in this section to install additional Adaptive Servers with SySAM in a network asset management system.

Before You Begin

Before you begin installing additional Adaptive Servers in a network asset management system, make sure you have the printed Sybase License Certificates that contain your Adaptive Server and options key information.

- Make sure you have installed Adaptive Server and the asset management software on the primary network node. See Chapter 3, "Unloading Server Products from Distribution Media".
- Make sure that the asset management software is running on the primary network node. See "Starting the SySAM" on page 43.
- Collect the following information:

- The host name of the primary network node.
- An unused port number on the primary network server node.
- The remote directory of the network server node on which the asset management software has been installed.
- The printed Sybase License Certificates for all of the optional features you have purchased for this server.

Starting the SySAM

If you are using networked software asset management system, you must run the software on the primary network node before installing additional servers.

Starting the Software Automatically

If you followed the unloading and licensing instructions in Chapter 3, "Unloading Server Products from Distribution Media", SySAM starts automatically if you choose to configure the server while in the Studio Installer.

Solaris Instructions

1 Make sure the *license.dat* file has the correct path for the **SYBASE** daemon.

2 Using an account with "root" privileges, log on to the machine where you installed the asset management software.

3 Use a text editor to create a new file, */etc/rc3.d/S17lmgrd*.

4 Add the following lines to */etc/rc3.d/S17lmgrd*:

```
SYBASE=<Sybase reldir>
SYBASE_ASE=ASE-12_0
/bin/su user_name -c 'echo starting lmgrd> \
   $SYBASE_SYSAM/bin/boot.log'

/bin/nohup /bin/su user_name -c 'umask 022; \
   $SYBASE_SYSAM/bin/lmgrd -c \
   $SYBASE_SYSAM/licenses/license.dat>> \
   $SYBASE_SYSAM/bin/boot.log'\

/bin/su user_name -c 'echo sleep 5>> \
   $SYBASE_SYSAM/bin/boot.log'

/bin/sleep 5
```

```
/bin/su user_name -c 'echo lmdiag >> \
   $SYBASE_SYSAM/bin/boot.log'

/bin/su user_name -c
'$SYBASE_SYSAM/bin/lmutil lmdiag -n -c \
   $SYBASE_SYSAM/bin/license.dat>> \
   $SYBASE_SYSAM/bin/boot.log'

/bin/su user_name -c 'echo exiting >> \
   $SYBASE_SYSAM/bin/boot.log'
```

5 Save */etc/rc3.d/S17lmgrd* and exit the text editor.

6 Change the permissions, ownership, and group for */etc/rc3.d/S17lmgrd* enter:

```
chmod 744 /etc/rc3.d/S17lmgrd
chown root /etc/rc3.d/S17lmgrd
chgrp sys /etc/rc3.d/S17lmgrd
```

Starting the Software Manually

If you did not use the SySAM utility to license Sybase server products during the unload process, you will need to start the asset management software manually:

- Use the UNIX command:

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

- Alternatively, run the *startd.sh* script in the *$SYBASE/$SYBASE_SYSAM/bin* directory

Verifying the Software is Running

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

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

Make sure that both **lmgrd** and **SYBASE** are running, i.e. “up” before you continue the installation or start Adaptive Server.

Enabling SySAM Features

Before enabling SySAM software, unload the server components to the client machine. See Chapter 3, “Unloading Server Products from Distribution Media” for installation instructions.

After the server components have been unloaded, follow these steps to enable SySAM features on client machines:

1 Verify that the licensing software is running. See“Verifying the Software is Running” on page 44.

2 After the components are copied, the installer prompts: “Do you have Sybase Software Asset Management Certificates to register?”

Click No. (The certificates should already be registered at the primary network node before installation of the Adaptive Server software on the enterprise client).

3 The installer prompts: “Have you registered Sybase Software Asset Management Certificates at a central license host?”

Click Yes.

4 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.

5 Click OK to continue the installation of Adaptive Server and PC-Client components.

See the Chapter 7, “Unloading Sybase PC-Client Products from Distribution Media” if you need help configuring installed components.

Adding Adaptive Server Features

To add features for which you have licenses:

1 Enable the new feature. See “Adding Additional Licenses” on page 46.

2 Shut down Adaptive Server. See Chapter 11, “Starting and Stopping Servers”.

3 Enter additional feature licenses by using the SySAM license manager, **lmgr**. See “Adding Additional Licenses” on page 46.

4 Reread the license file. See “Adding Additional Licenses” on page 46.

5 Restart Adaptive Server. See Chapter 11, “Starting and Stopping Servers”.

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

Adding Additional Licenses

If you install additional components that require licenses, you need to update your license information.

To add additional licenses to SySAM:

1 Log on to the primary network node (server) that hosts the SySAM services.

2 Check to see that **lmgrd** and Sybase daemons are running:

```
$SYBASE/$SYBASE_SYSAM/bin/lmutil lmstat -c
```

3 If the license daemon is not running, manually start **lmgrd**:

```
$SYBASE/$SYBASE_SYSAM/bin/
lmgrd -c $SYBASE/$SYSAM/licenses/license.dat \
-l $SYBASE/$SYSAM/lmgrd.log&
```

4 Launch the license manager:

```
$SYBASE/SYSAM-1_0/bin/lmgr
```

5 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

Click More until you have entered all available licenses. Click Done.

6 Because you are adding additional licenses to an existing file, you must notify the license daemons of the changes. Before you issue the reread command:

- Verify that SySAM is running:

  $SYBASE/$SYBASE_SYSAM/bin/lmutil lmstat -c

- Shut down the server to which you are adding the new license. See Chapter 11, “Starting and Stopping Servers”.

7 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.

8 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.

Adding a License to a Pre-Existing Server

To upgrade a base-level Adaptive Server to a fully licensed product, you must update license information and reread the license file.

1 Check to see that **lmgrd** and Sybase daemons are running:

```
$SYBASE/$SYBASE_SYSAM/bin/lmutil lmstat -c
```

2 If the license daemon is not running, start the **lmgrd**:

```
lmgrd -c $SYBASE/$SYBASE_SYSAM/license.dat \
-l $SYBASE$SYBASE_SYSAM/log/lmgrd.log&
```

3 Run **lmgr**:

```
$SYBASE/SYSAM-1_0/bin/lmgrd
```

4 Provide the license information.

- Order Number
- Feature Name
- Feature Count
- Software Version

- Authorization Code

5 Once the license has been added, run the license management utility **lmutil lmreread** in the *$SYBASE/SYSAM-1_0/bin* directory. Enter the command

```
lmutil lmreread
```

The new licenses are appended to the license file.

6 If necessary, check the *lmgrd.log* file if you encounter problems.

$SYBASE/$SYBASE_SYSAM/log/lmgrd.log

7 If you encounter problems with new licenses, check the *lmgrd.log* file in the *%SYBASE%\%SYBASE_SYSAM%\bin* directory to see that they were properly appended to the license file.

Starting Adaptive Server with SySAM features

When you start Adaptive Server, it automatically tries to check out a license for ASE_SERVER. Adaptive Server attempts to check out licenses for each configured optional feature.

Optional features, such as distributed transaction management, can be turned on or off using **sp_configure** parameters. Table 4-3 lists the parameters used to configure optional features.

Table 4-3: Configuration parameters for optional features

Feature Name	License Name	Configuration Parameter
High Availability	ASE_HA	enable HA
Distributed Transaction Management	ASE_DTM	enable xact coordination enable DTM
Java in Adaptive Server	ASE_JAVA	enable java
Advanced Security Mechanisms	ASE_ASM	use security services

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 reboot Adaptive Server. During the boot sequence, Adaptive Server attempts to check out licenses for all configured features.

Note Some features require additional preparation before you can use them. For more information, see Chapter 10, "Adding Optional Functionality to Adaptive Server".

- 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.*

Troubleshooting SySAM

When you boot 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_ASE/$SYBASE_SYSAM/log*).

Table 4-4 describes the SySAM error messages and suggests ways to correct common problems.

Table 4-4: SySAM error messages

Message	Description	Action
Use license file *filename*.	This message identifies the license file that Adaptive Server uses.	This is a normal start-up message; no action is required.
Feature *feature_name* is not licensed.	There is no license available for the specified optional feature.	You must purchase and install a valid license for the feature before it can be used. See Chapter 3, "Unloading Server Products from Distribution Media".

Message	Description	Action
There is no valid license for ASE server product. Server is booting with all the optional features disabled.	Adaptive Server cannot find a valid license file, or it cannot contact the asset management software to check out an ASE_SERVER license.	When Adaptive Server cannot check out an ASE_SERVER license, the server boots but does not enable any optional features. If you have purchased an ASE_SERVER license, make sure it has been installed. Also, make sure the license file exists and has been defined in LM_LICENSE_FILE.
License manager initialization fails.	The asset management software did not initialize.	Use "Starting the SySAM" on page 43.
Feature *feature_name* is over drafted *n* times.	All licenses for the specified feature have already been checked out by other servers.	Adaptive Server enables the optional feature, even though all available licenses have been used by other servers. Contact your authorized Sybase reseller to purchase an additional license for the feature.
Warning: there is no valid license for ASE_JAVA and therefore feature JAVA is not initialized.	There is no license available for the configured feature.	Adaptive Server does not enable the optional feature until it can obtain a valid license. Purchase and install a license for the feature. Contact your authorized Sybase reseller.
Warning: there is no valid license for ASE_ASM and therefore feature ASM is not initialized.	There is no license available for the configured feature.	Adaptive Server does not enable the optional feature until it can obtain a valid license. Purchase and install a license for the feature. Contact your authorized Sybase reseller.
Warning: ASE_HA does not have a valid license and therefore is not initialized.	There is no license available for the configured feature.	Adaptive Server does not enable the optional feature until it can obtain a valid license. Purchase and install a license for the feature. Contact your authorized Sybase reseller.
Warning: ASE_DTM does not have a valid license and therefore is not initialized.	There is no license available for the configured feature.	Adaptive Server does not enable the optional feature until it can obtain a valid license. Purchase and install a license for the feature. Contact your authorized Sybase reseller.

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 12.0 PERMANENT 1000
123456789123 SN=10001 OVERDRAFT=10000 ck=0

INCREMENT ASE_JAVA SYBASE 12.0 PERMANENT 1000
123456789123 SN=10001 OVERDRAFT=10000 ck=0

INCREMENT ASE_DTM SYBASE 12.0 PERMANENT 1000
123456789123 SN=10001 OVERDRAFT=10000 ck=0

INCREMENT ASE_HA SYBASE 12.0 PERMANENT 1000
123456789123 SN=10001 OVERDRAFT=10000 ck=0
```

- *ASE_<FEATURE>* is the feature name, such as ASE_SERVER, ASE_JAVA, etc.
- *12.0* 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.

What's Next?

Some references that may help you get started using your new Adaptive Server are:

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

For information on developing a backup and recovery plan and backing up and restoring user databases, see the *System Administration Guide*.

Monitor Server requires some additional configuration after installation. See the *Adaptive Server Enterprise Monitor Server User's Guide*.

For information on using extended stored procedures, see the *Transact-SQL User's Guide*.

For information on expanding the functionality to your Adaptive Server, see Chapter 10, "Adding Optional Functionality to Adaptive Server".

For information on localizing your Adaptive Server, see Chapter 8, "Customizing Localization for Adaptive Server".

CHAPTER 5

Installing Sybase Servers

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

Topics covered include:

Name	Page
Overview	53
Pre-Installation Tasks	54
Determining the Location, Type, and Size of a Database Device	56
Installing Server Components	58
Installation Methods	58
Setting Environment Variables	59
Changing the Default Language and Character Set	72
Resource File Installation	73
What's Next?	77

Overview

After you have unloaded the product files, you are ready to install one or more of the following Sybase servers:

- Adaptive Server
- Backup Server
- Full-Text Search (FTS) engine
- Monitor Server

The basic process for installing Sybase servers is:

1 Perform the pre-installation tasks as described in "Pre-Installation Tasks" on page 54.

2 Perform the installation using any of the methods described in "Installation Methods" on page 58.

3 Perform the post-installation tasks as described in Chapter 6, “Post-Installation Tasks”.

Pre-Installation Tasks

Before installing the Sybase servers:

1 Read the “Special Installation Instructions” section in the release bulletin.

2 Verify that your operating system meets the release-level, RAM, and network-protocol requirements.

These requirements are fully documented in Chapter 2, “Installation Requirements” and in the release bulletin.

3 Install operating system patches, if required.

Required operating system patches are described in the release bulletin for the product.

4 Verify that your network is configured to run in loopback mode.

If Adaptive Server and Sybase client applications are installed on a machine that is not connected to a network, Sybase software still uses the network software, which must be configured to run in loopback mode.

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 again, using the same command.

c Log out twice.

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.

5 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 memory** configuration parameter, which defaults to the values shown in Table 5-1.

Table 5-1: Shared memory requirements

Platform	Number of 2K Pages	total memory Parameter Default
Sun Solaris	(32,768) 32-bit [47,104] 64-bit	(64MB) [92MB]

To adjust the shared memory value of the operating system, add the following line to the operating system configuration file */etc/system*:

```
set shmsys:shminfo_shmmax = nnn
```

where *nnn*, set in bytes, is at least 32MB for a default Adaptive Server. Set this value higher if you plan to increase Adaptive Server's total memory to more than the default value shown in Table 5-1.

Note Reboot your machine to activate the changes made to the operating system configuration file.

After you install Adaptive Server, you can change the total memory parameter by using sp_configure.

6 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.

To adjust shared memory segments of the operating system, add the following line to the configuration file */etc/system*:

```
set shmsys:shminfo_shmseg=x
```

where *x* is the number of shared memory segments.

7 Enable asynchronous disk I/O.

8 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 56.

9 Review the SySAM procedures and plan your SySAM configuration using Chapter 4, "Sybase Software Asset Management (SySAM)".

Determining the Location, Type, and Size of a Database Device

Adaptive Server requires several database devices. Table 5-2 below shows baseline values for each of the devices. See the release bulletin for any last-minute changes to these values.

Table 5-2: Adaptive Server database devices

Device	Purpose	Minimum Size	Recommended Size
master	Stores system databases	25MB	30MB
sysprocsdev (also called the *sybsystemprocs* device)	Stores the *sybsystemprocs* database	80MB	80MB (plus any space for holding stored procedures that you have created)
sybsystemdb	Transaction processing	4MB	5 – 20MB
sybsecurity (optional)	Required for auditing	0–5MB	7MB; more for specialized auditing

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

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

Do not run Adaptive Server on network-mounted devices, such as NFS- or RFS-mounted files, for these reasons:

- Network problems can cause serious data integrity problems because the network lies between Adaptive Server and the devices on which it writes.

- If the remote machine is down when you reboot Adaptive Server, and database devices cannot be activated, automatic recovery fails.
- Adaptive Server activity is suspended when the remote machine or network is down.

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

To determine whether a device is a block device or a character device:

Run **ls -l** on the */dev* directory.

In the left-most column, “b” (block) or “c” (character) appears.

To choose 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 5-2 on page 56.

4 Verify that the device is a character device.

5 Verify with the operating system administrator that the partition you have chosen is available.

6 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.

7 If you have not unloaded the Adaptive Server software from the CD onto your system, complete the procedures in Chapter 3, “Unloading Server Products from Distribution Media”.

Installing Server Components

This section describes how to install Adaptive Server version 12.0. Follow the instructions in this section to install Adaptive Server, Backup Server, or Monitor Server.

Be sure that you are logged in as “sybase”, and that you have performed the “Pre-Installation Tasks” on page 54.

Installation Methods

Use any of the following methods to install Sybase Servers:

- Studio Installer – use the Studio Installer to calls the installer utility to install servers and customize them for a production environment. Fully customizing a server at installation time reduces the need to make changes later. The Studio Installer is described in Chapter 3, “Unloading from Distribution Media”.
- Resource file – First use the Studio Installer to unload the server components, then run the **srvbuildres** utility to install Adaptive Server and Backup Server for additional sites that require identical servers. 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 directly 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.

Setting Environment Variables

After exiting the Studio Installer, you must set some environment variables before you can configure and run Adaptive Server and other components successfully.

Studio Installer unloads C shell and Bourne shell files containing environment variables. The files locations are:

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

To set the environment variables, you can either:

- Modify the component's RUN environment permanently using the environment variable values from the shell files, or
- Update the component's RUN environment interactively, before invoking the component.

To update the RUN environment immediately:

1 Change the window to a Bourne shell by entering **sh**.

2 Enter:

```
.SYBASE.sh
```

3 Change the window back to a C shell by entering **csh**.

4 Alternatively, you may source the *SYBASE.csh* using the command: **source** SYBASE.csh.

For more information about setting environment variables, see Chapter 13, "Configuring the Operating System".

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

Installation

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

1 Be sure that you are logged in as "sybase".

2 Source the SYBASE.sh script files to set the environment variables.

3 Then use the **asecfg** utility to call one of the following:

- **srvbuild** utility to build a server,
- **sqlupgrade** to upgrade Sybase servers,
- **sqlloc** to localize the server, or
- **dsedit** to edit the interfaces file.

4 To start **asecfg,** at the command line enter:

```
$SYBASE/ASE-12_0/bin/asecfg
```

5 Then click on the Build New Server. The **srvbuild** screen is displayed.

6 Choose the server type that you want to configure.

Only the server types that were unloaded from the distribution media are displayed on the **srvbuild** screen. For example, if you did not unload Monitor Server, it does not appear on the **srvbuild** screen.

If you unloaded Monitor Server for SQL Server 11.0.x, it does not appear on the **srvbuild** screen. For information on how to install and configure this product, see the release bulletin for Adaptive Server Enterprise Monitor.

If this is a first-time installation for your site, you must select Adaptive Server. You can choose all server types or any combination of server types in addition to Adaptive Server. **srvbuild** consecutively displays the attribute screen for each server you choose, in the order listed on the screen.

7 Provide a name for each server you install.

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.

If you are installing a new Backup Server to accompany an upgraded Adaptive Server, follow these guidelines when choosing your Backup Server name:

- If you are upgrading from version 4.9.x, choose a name that is unique to your site.
- If you are upgrading from version 10.x or 11.x, do not change the name of Backup Server.

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

8 Click OK.

9 Go to the section in this manual that describes the next part of the custom installation process for the product you chose to install.

If you chose:

- Adaptive Server – go to "Installation – Adaptive Server" on page 61.
- Backup Server – go to "Installation – Backup Server" on page 64.
- FTS – see the *Standard Full-Text Search Specialty Data Store User's Guide* manual.
- Monitor Server – go to "Installation – Monitor Server" on page 67.

Installation – Adaptive Server

The following procedure assumes that you have completed the procedures in the section "Pre-Installation Tasks" on page 54. To continue installing Adaptive Server, complete the following procedures.

If you selected Adaptive Server to be installed, the Adaptive Server attribute screen is displayed.

1 Provide a location for the master device. The default directory is: *$SYBASE*.

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
```

where *disk_partition* is the disk partition to use.

2 Provide a size for the master device.

The default value, 30MB, is the recommended size.

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, you can choose the size. 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.

3 Provide a size for the *master* database.

The default size is recommended. 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 tables.

4 Specify a disk partition or a file name for the *sybsystemprocs* device.

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

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.

---

5 Provide a size for the *sybsystemprocs* device.

The default value, 80MB, is the recommended size. Because the *sybsystemprocs* database is constantly increasing in size, this allows enough space on the device for growth.

If you entered a disk partition as the *sybsystemprocs* device, the *sybsystemprocs* device size reflects the size of the disk partition. You do not need to change the size.

If you entered an operating system file, you can change the size. You can provide a size that is less than the default size, but the minimum allowable size is 80MB.

If you entered an operating system file, you can change the size. You can provide a size that is less than the default size, but the minimum allowable size is 80MB. The maximum size is the size of the file system.

6 Provide an error log path name.

The default is:

$SYBASE/ASE-12_0/install/errorlog

You can provide a path name other than the default path provided.

7 For Interfaces File Entry, specify the transport type, host name, and port number.

These entities comprise the interfaces file entry which defines 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 installing Adaptive Server.

  If you change the default port number, verify that the new port 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.

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

- Specify a *sybsystemdb* device path for the *sybsystemdb* database. See Chapter 10, “Adding Optional Functionality to Adaptive Server”.

- Place shared memory files in a directory other than the default location under *$SYBASE*.
- Change your default Backup Server.

9 To create the server, click Build Server.

srvbuild displays a Status Output screen that shows the status of the various installation tasks as they execute. Go to "Viewing the Installation Status Output Screen" on page 71.

10 Do one of the following:

- Proceed to the attribute screen for the next server type. Then, go to the section in this manual that describes the next part of the custom installation process for that server. If you chose:

  Backup Server – go to "Installation – Backup Server" on page 64.

  Monitor Server – go to "Installation – Monitor Server" on page 67.

  FTS – go to "Installation – FTS" on page 70.
- Verify that the installation was successful as described in Chapter 6, "Post-Installation Tasks"
- Complete the post-installation tasks as described in Chapter 6, "Post-Installation Tasks".

Installation – Backup Server

The following procedure assumes that you have completed the procedure in the section "Pre-Installation Tasks" on page 54. To continue installing Backup Server, complete the following procedure.

1 If you selected Backup Server to be installed, the Backup Server attribute screen is displayed.

2 Provide the related Adaptive Server name. This is the name of the Adaptive Server that will use this Backup Server.

If the related Adaptive Server is created in the same **srvbuild** session as Backup Server, this field does not appear. The Adaptive Server you are creating is used as the related Adaptive Server.

3 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.

4 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.

5 Determine the error log path name for Backup Server. Accept the default provided or provide another location.

6 Determine the location of 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.

The default location is *$SYBASE/$SYBASE_ASE/backup_tape.cfg*.

7 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 downloaded from distribution media to your system.

By default, all character sets are copied from the distribution media when performing a Standard or Full install.

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

The default character set is iso_1.

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

9 Specify the maximum number of network connections.

You can accept the default or change it.

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

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

10 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 (backup) operation
- One connection for each load (recovery) operation
- One connection for volume change messages

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

11 For the Interfaces File Entry attribute, 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 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 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.

12 Do one of the following:

- Proceed to the attribute screen for the next server type. Then, go to the section in this manual that describes the next part of the custom installation process for that server. If you chose:

  Monitor Server – go to "Installation – Monitor Server" on page 67.

  FTS – go to "Installation – FTS" on page 70.

- To create only the server or servers you have specified, click Build Server.

  srvbuild displays a Status Output screen that shows the status of the various installation tasks as they execute. Go to "Viewing the Installation Status Output Screen" on page 71.

Installation – Monitor Server

The following procedure assumes that you have completed the procedure in the section "Pre-Installation Tasks" on page 54. To continue installing Monitor Server, complete the following procedure.

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

2 Provide the related Adaptive Server name.

  This is the name of the Adaptive Server that Monitor Server will be monitoring. This field does not appear if the related Adaptive Server is created in the same **srvbuild** session as Monitor Server. The Adaptive Server you are creating is used as the related Adaptive Server.

3 Specify the Sybase System Administrator (SA) user name.

  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.

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

4 Provide the Adaptive Server SA password.

This is the password of the Monitor Server superuser.

If the password is omitted, Monitor Server prompts for a password during start-up.

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

5 Determine the maximum number of connections allowed to Monitor Server.

Valid values are 1–20. The default is 5.

A connection to Monitor Server can be any of the following:

- 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 the default configuration of five connections is in effect, three additional connections to Monitor Server are 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 *ms.log*.

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 *servername* is the name of your Adaptive Server.

This specifies the name of the directory where the *servername.krg* shared memory file is located.

If this parameter is omitted, Monitor Server looks for the shared memory $SYBASE 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 sever.

9 For the Interfaces File Entry attribute, 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.

  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 Do one of the following:

- Proceed to the attribute screen for the next server type. Then, go to the section in this manual that describes the next part of the custom installation process for that server. If you chose:

  FTS – go to “Installation – FTS” on page 70.

- To create only the server or servers you have specified, click Build Server.

  srvbuild displays a Status Output screen that shows the status of the various installation tasks as they execute. Go to "Viewing the Installation Status Output Screen" on page 71.

Installation – FTS

The following procedure assumes that you have completed the procedure in the section "Pre-Installation Tasks" on page 54. To continue installing FTS, complete the following procedure.

1 Errorlog: The default is *$SYBASE/$SYBASE_FTS/install/<servername>.log*

2 Collection directory: Accept the default, *$SYBASE/SYBASE_FTS/collection*s

3 Default Database: text_db

4 Language: us_english

5 Character set: iso_1

6 Minimum number of sessions: 10

7 Maximum number of sessions: 100

8 For the Interfaces File Entry attribute, specify the transport type, host name, and port number. These entities comprise the interfaces file entry. This entry enables FTS 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 name of the machine where you installed 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 installing the FTS.

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

- Adaptive
- FTS

9 To create the server, click Build Server.

srvbuild displays a Status Output screen that shows the status of the various installation tasks as they execute. Go to "Viewing the Installation Status Output Screen" on page 71.

See *Standard Full-Text Search Specialty Data Store User's Guide* for information on the FTS feature.

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 approximately 10 to 15 minutes to build each server.

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

If the installation is not successful, see Chapter 12, "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.
```

2 If you installed Adaptive Server during this **srvbuild** session, go to "Changing the Default Language and Character Set" on page 72.

3 If you installed Adaptive Server during this **sybatch** session, go to "Changing the Default Language and Character Set" on page 72.

Changing the Default Language and Character Set

Warning! Make all changes to the 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 may require additional steps. To change the character set or sort order after you have added data, see the *System Administration Guide*.

If you installed Adaptive Server, **srvbuild** displays a message box asking if you want to localize your Adaptive Server to a language other than us_english and if you want to use a character set or sort order other than the default.

srvbuild creates an Adaptive Server with the following defaults:

- us_english language
- iso_1 character set
- roman8 character set
- Binary sort order

Valid language options depend on what language modules were unloaded from the distribution media onto your system.

All character sets are copied from the distribution media by default.

You can:

- Click No to accept the defaults, and then go to "Installation Complete" on page 72.
- Click Yes to change the defaults.

  The **sqlloc** menu is displayed. **sqlloc** is the GUI utility used to change default languages, character sets, and sort orders. When you finish this task, go to "Installation Complete" on page 72.

  For information on using **sqlloc**, see Chapter 8, "Customizing Localization for Adaptive Server".

Installation Complete

You have completed the installation process. Go to Chapter 6, "Post-Installation Tasks".

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 3, "Unloading Server Products from Distribution Media".

Note You cannot install Monitor Server, XP Server or FTS using resource files.

To install an Adaptive Server or Backup Server using resource files:

1 Edit a resource file as described in "Editing a Resource File" on page 73.

2 Execute the **srvbuildres** utility, using the edited resource file as described in "Using srvbuildres" on page 77.

Editing a Resource File

You can edit a resource file by:

- Editing a sample resource file
- Editing the resource file 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 "_back" extensions for Backup Server. For example, the Backup Server for PIANO should be PIANO_back.
- The master device size must be at least 30MB.
- The system procedure device and *sybsystemprocs* database sizes must be at least 80MB.
- The auditing entries in the resource file are not supported and will not be processed by the utility. To establish auditing, use the procedures provided in Chapter 10, "Adding Optional Functionality to Adaptive Server".

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 or Backup servers 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.

Verifying That You Can Connect to Servers

Use **isql** or **dsedit** to perform a quick test.

1 Use **isql** to connect to servers:

- 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.

- To display the Adaptive Server version number, enter:

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

  Adaptive Servers's version number is displayed. The output should show Adaptive Server at version 12.0.

Resource File Attributes for Adaptive Server

Table 5-3 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 5-3 are the attribute names provided in the sample resource file included in your Adaptive Server distribution.

Table 5-3: 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	80 (MB)
sqlsrv.sybsystemprocs_database_size	80 (MB)
sqlsrv.sybsytemdb_device_physical_name	*path_and_name_of_ sybsystemdb_device*

Attribute	Default Value [Other Options]
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 5-4 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 5-4 are the attribute names provided in the sample resource file included in your Adaptive Server distribution.

Table 5-4: Resource file attributes for Backup Server

Attribute	Default Value [Other Options]
sybinit.release_directory	$SYBASE
sybinit.product	bsrv
bsrv.server_name	*server_name*_back
bsrv.do_add_backup_server (to interfaces file)	yes [no]
bsrv.network_protocol_list	tcp [spx]
bserv.network_hostname_list	*hostname*
bsrv.network_port_list	*port_number*
bsrv.language	us_english [chinese, french, german, japanese, spanish]
bsrv.character_set	iso_1[iso_1, cp850, cp437, deckanji, ascii_8, eucgb, eucjis, mac, roman8, sjis, utf8]
bsrv.tape_config_file	*$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 6, “Post-Installation Tasks”.

What’s Next?

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:

	• *Introducing Sybase Adaptive Server for UNIX Platforms* for an overview of Adaptive Server • "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*.
FTS	For information on using Full-Text Search capabilities, see the *Standard Full-Text Search Specialty Data Store User's Guide.*

CHAPTER 6

Post-Installation Tasks

This chapter discusses post-installation tasks, including the configuration of jConnect, and describes the method for uninstalling servers. This chapter covers these topics:

Name	Page
Post-Installation Tasks	79
Resetting Variables	80
Setting the Sybase System Administrator Password	79
Verifying That Servers Are Running	80
Setting jConnect Environment Variables	81
Installing Stored Procedures for jConnect	83
What's Next?	89

Post-Installation Tasks

After completing the custom installation:

- Setting the System Administrator's Password
- Resetting Variables

Setting the Sybase System Administrator Password

A user account called "sa" is created for the Sybase System Administrator when you install the Sybase software. See "User Roles" on page 2. 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
```

Resetting Variables

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, TEMP, CLASSPATH, and SYBASE.

Verifying That Servers Are Running

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

- 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>
  ```

- At the UNIX prompt, enter:

  ```
  $SYBASE/$SYBASE_ASE/install/showserver
  ```

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

Information Required for XP Server

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
```

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

Setting jConnect Environment Variables

To test the jConnect installation and use the jConnect classes you installed, you must set the JDBC_HOME and CLASSPATH environment variables.

Setting JDBC_HOME

JDBC_HOME represents the directory where you install jConnect.

Note The ZIP file installs everything below a directory with the specific jConnect driver's name.

For example, if you unloaded *jconnect-5_2*, you would set JDBC_HOME to:

```
$SYBASE/jConnect-5_2
```

Setting CLASSPATH

CLASSPATH is the location of the jConnect runtime classes and/or JAR files, and the Java home library for JDK 1.1.x. This section describes what you need in your CLASSPATH depending on which versions of JDK and jConnect you are using.

Note If you are using JRE or jView instead of JDK, see the documentation for those products for instructions on setting the CLASSPATH variable.

jConnect 4.x and JDK 1.1.x

To use jConnect 4.x with JDK 1.1.x, set the CLASSPATH to:

$JAVA_HOME/lib/classes.zip;$JDBC_HOME/classes

jConnect 4.x and Java 2 (JDK 1.2)

To use jConnect 4.x with Java 2, set the CLASSPATH to:

$JDBC_HOME/classes

Java 2 (JDK 1.2) does not require you to set the location of the Java runtime classes in your CLASSPATH.

jConnect 5.x and Java 2 (JDK 1.2)

To use Java 2 (JDK 1.2) with jConnect 5.x, set the CLASSPATH to:

$JDBC_HOME/classes/jconn2.jar

Java 2 (JDK 1.2) does not require you to set the location of the Java runtime classes in your CLASSPATH.

To run the samples or the TDS-tunnelling servlet, set the CLASSPATH to:

$JDBC_HOME/classes/jconn2.jar;$JDBC_HOME/classes

Note If you install both jConnect 4.x and jConnect 5.x, be sure that your CLASSPATH is set correctly. Sybase recommends that you do not run JDK 1.1.x with jConnect 5.x

In addition, the CLASSPATH should include the location of any application classes you have defined.

As explained in the next section, if you use Netscape, some restrictions may apply to the way you set your CLASSPATH environment variable.

Restrictions on Setting CLASSPATH When Using Netscape

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.

Installing Stored Procedures for jConnect

To use function escapes and **DatabaseMetaData** methods, you must install stored procedures on the Adaptive Server Enterprise or Adaptive Server Anywhere database against which you want to use these methods. These stored procedures are also required by some of the connection methods.

jConnect provides scripts to install the required stored procedures and tables.

- *sql_server.sql* installs stored procedures on pre-version 12.0 Adaptive Server Enterprise databases.
- *sql_server12.sql* installs stored procedures on an Adaptive Server Enterprise version 12.0 databases.
- *sql_anywhere.sql* installs stored procedures on SQL Anywhere and Adaptive Server Anywhere databases.

To run the scripts:

1 Make sure that your JAVA_HOME, JDBC_HOME, and CLASSPATH environment variables are set.

2 To connect to a SQL Anywhere database earlier than version 6.0, start an Open Server gateway.

3 Use the **IsqlApp** sample application according to the instructions in the following sections, based on the target database and your platform.

Adaptive Server Enterprise

To install stored procedures for metadata in an Adaptive Server Enterprise database on UNIX, enter the following command from your JDBC_HOME/classes directory:

ASE version 12.0:

```
java IsqlApp -U sa -P password -S jdbc:sybase:
   Tds:[hostname]:[port] -I $JDBC_HOME/sp/sql_server12.sql -c go
```

ASE version 11.9.2 and earlier:

```
java IsqlApp -U sa -P password -S jdbc:sybase:
   Tds:[hostname]:[port] -I $JDBC_HOME/sp/sql_server.sql -c go
```

SQL Anywhere or Adaptive Server Anywhere

To install stored procedures for metadata in a SQL Anywhere or Adaptive Server Anywhere database on a UNIX system, enter the following command on one line from your *JDBC_HOME/classes* directory:

```
java IsqlApp -U dba -P password -S jdbc:sybase:
   Tds:[hostname]:[port] -I $JDBC_HOME/sp/sql_anywhere.sql -c go
```

jConnect Version 4.2 and 5.2

Reviewing Installation Contents

Once you complete the jConnect installation, review the contents listed in this section.

Packages

jConnect consists of the following packages in the jConnect installation directory.

jConnect 4.2:

- **com.sybase.jdbc**
- **com.sybase.jdbcx**
- **com.sybase.tds**
- **com.sybase.timedio**
- **com.sybase.utils**

The jConnect driver, **com.sybase.jdbc.SybDriver**, is in the **com.sybase.jdbc** package.

jConnect 5.2:

For jConnect version 5.2, the following packages are found inside JAR files: *jconn2.jar* for the release build, *jconn2d.jar* for the debug build.

- **com.sybase.jdbc2.jdbc**
- **com.sybase.jdbc2.tds**
- **com.sybase.jdbc2.timedio**
- **com.sybase.jdbc2.utils**
- **com.sybase.jdbcx**

The jConnect driver, **com.sybase.jdbc2.jdbc.SybDriver**, is in the **com.sybase.jdbc2.jdbc** package.

Directories and Files

Table 6-1 list the directories and files created when you install jConnect version 4.2 and the jConnect documentation.

Table 6-2 list the directories and files created when you install jConnect version 5.2 and the jConnect documentation.

Table 6-3 lists the directories and files created when you install the jConnect free utilities.

Table 6-1: jConnect 4.2 installation

Name	Type	Description
jconnect-4_2	Directory	Contains all of the jConnect version 4.2 files and subdirectories.

Name	Type	Description
jConnect_docs	Directory	Contains the */docs* subdirectory, which contains the *Programmer's Reference* in HTML and PDF format.
/classes	Subdirectory	The following jConnect 4.2 components are installed in the */classes* subdirectory: • *gateway* subdirectory for compiled TDS-tunnelling servlet • *com* subdirectory for jConnect 4.2 classes. • *sample* subdirectory with jConnect 4.2 class files for sample applets and applications, including the Isql applet.
/devclasses	Subdirectory	*/com* subdirectory for jConnect 4.2 classes with debug mode turned on.
/docs	Subdirectory	Contains the */en* subdirectory, which contains the English javadoc documentation.
/gateway	Subdirectory	Contains source code for the TDS-tunnelling servlet.
/sample	Subdirectory	Contains source code for sample Java applications.
/sp	Subdirectory	Contains isql scripts that install stored procedures for function escapes and DatabaseMetaData methods on a database server. The scripts are: • sql_anywhere.sql • sql_server12.sql (ASE version 12.0) • sql_server.sql (pre-version 12.0 ASE)
/tools	Subdirectory	Contains a perl script (*decode-tli*) that UNIX users can apply to parse an *interfaces* file into a readable format.
index.html	HTML File	Contains links to jConnect documentation and samples.

Table 6-2: jConnect 5.2 installation

Name	Type	Description
jconnect-5_2	Directory	Contains all of the jConnect version 5.2 files and subdirectories.
jConnect_docs	Directory	Contains the */docs* subdirectory, which contains the *Programmer's Reference* in HTML and PDF format.
/classes	Subdirectory	The following jConnect 5.2 components are installed in the */classes* subdirectory: • *jconn2.jar* file that contains jConnect 5.2 classes • *sample2* subdirectory with jConnect 5.2 class files for sample applets and applications, including the Isql applet • *gateway2* subdirectory for compiled TDS-tunnelling servlet
/devclasses	Subdirectory	*jconn2d.jar* file that contains the same jConnect 5.2 components as *jconn2.jar*, but debug mode is turned on.
/docs	Subdirectory	Contains the */en* subdirectory, which contains the English javadoc documentation.

Name	Type	Description
/gateway2	Subdirectory	Contains source code for the TDS-tunnelling servlet.
/sample2	Subdirectory	Contains source code for sample Java applications.
/sp	Subdirectory	Contains isql scripts that install stored procedures for function escapes and DatabaseMetaData methods on a database server. The scripts are: • sql_anywhere.sql • sql_server12.sql (ASE version 12.0) • sql_server.sql (pre-version 12.0 ASE)
/tools	Subdirectory	Contains a perl script (*decode-tli*) that UNIX users can apply to parse an *interfaces* file into a readable format.
index.html	HTML File	Contains links to jConnect documentation and samples.

Table 6-3: Free utilities installation

Name	Type	Description
jutils-2.0	Directory	Contains all of the utility files and subdirectories listed below.
/cascade	Subdirectory	Contains */uk* subdirectory with the source code and compiled code for the Cascade gateway. For more information, see at http://www.cascade.org.uk.
/jisql	Subdirectory	Contains the jisql program and the */doc* and */HelpFiles* subdirectories.
/ribo	Subdirectory	Contains the Ribo program and the */doc* subdirectory.

JDBC 1.x Samples

You can find JDBC 1.x code samples in the *sample* subdirectory under the jConnect installation directory. See “Running jConnect Sample Programs and Code,” in Appendix B of the *Sybase jConnect for JDBC Programmer's Reference* for more information.

JDBC 2.0 Samples

You can find JDBC 2.0 code samples in the *sample2* subdirectory under the jConnect installation directory. See “Running jConnect Sample Programs and Code,” in Appendix B of the *Sybase jConnect for JDBC Programmer's Reference* for more information.

Testing the Installation

Once jConnect is installed, test the installation by running the **Version** program.

The **Version** program connects to a demonstration database that Sybase has made available on the Internet. You must have Internet access to run the **Version** program successfully, or explicitly run it with **[-U *username*] [-P *password*] [-S *servername*]** commands to point it to your database.

From the DOS prompt for Windows NT and Windows 95, or at a UNIX prompt, change to the JDBC_HOME directory and enter the following command.

jConnect 4.2:

```
java sample.SybSample Version
```

jConnect 5.2:

```
java sample2.SybSample Version
```

This should bring up a SybSample window. At the top of the window, the Running Sybase Sample text box should display the source code for **Version** as the program runs. The middle text box—Sample Output—should display version information. For example:

```
Using JDBC driver version 4.2
jConnect (TM) for JDBC(TM)/4.2...
```

or

```
Using JDBC driver version 5.2
jConnect (TM) for JDBC(TM)/5.2...
```

- If the message above appears in the Sample Output text box, you have successfully installed jConnect.
- If you get the above message, but the source code for **Version** does not appear in the Running Sybase Sample text box, and the Status text box at the bottom of the window shows:

  ```
  java.io.FileNotFoundException: Version.java
  ```

  you successfully installed jConnect, but probably did not enter the command to run the **Version** program from your *JDBC_HOME* or *JDBC_HOME/sample* (for jConnect 4.2) or *JDBC_HOME/sample*2 (for jConnect 5.2) directory.
- If the SybSample window does not appear and you see this error message:

  jConnect 4.2:

  ```
  Can’t find class sample.SybSample
  ```

  jConnect 5.2:

  ```
  Can’t find class sample2.SybSample
  ```

  check your CLASSPATH to be sure that it is set correctly.

What's Next?

- To configure the server and utilities for non-English language operations, you must install the appropriate language module and perform the configuration. See Chapter 8, "Customizing Localization for Adaptive Server".
- To install additional Sybase products, see Chapter 5, "Installing Sybase Servers".
- To add optional functionality, see Chapter 10, "Adding Optional Functionality to Adaptive Server".
- To start and stop Adaptive Server, see Chapter 11, "Starting and Stopping Servers".

CHAPTER 7

Unloading Sybase PC-Client Products from Distribution Media

This chapter describes how to unload and install additional Adaptive Server products, such as Sybase Central and client plug-ins. This chapter covers the following topics:

Name	Page
Overview	91
Product Descriptions	91
Unloading PC-Client Products	94
Installing Open Client/C	101
Installing InfoMaker	103
Configuring Network Connections for Client Products	104
Enabling TCP Connections	106
Installing the Entire Java Development Kit	98
Using Open Database Connectivity	107
What's Next?	107

Overview

Client plug-ins enable clients, such as Sybase Central and Open Client, to access Adaptive Server. See the release bulletin for a complete list of PC-Client components.

Product Descriptions

Table 7-1 describes Adaptive Server client products, which can be installed on Windows 95, Windows 98, and Windows NT client computers.

Table 7-1: Products and platforms

Product and Platforms	Description
Open Client/Server version 12.0	Used to develop and deploy C language-based applications that access Adaptive Server data. Default installation includes: • Client-Library™ • DB-Library • CS-Library • Bulk Library • bcp utility used to transfer data between files and Adaptive Server databases • defncopy utility used to copy stored procedure definitions from one database to another • Online help files for Open Client routines • Net-Library drivers for connecting 32-bit client applications to a server through most available network protocols. Default installation includes: dsedit for editing the *sql.ini* interfaces file and testing client-server connections Drivers for Named Pipes and Windows Sockets Drivers for Named Pipes, Windows Sockets, Novell LAN Workplace, 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: Novell SPX/IPX, DEC PathWorks, PC-NFS TCP/IP, FTP PC/TCP, Wollongong PathWay TCP/IP, and Net Manage TCP/IP NWLink SPX/IPX, DEC PathWorks, and debug support, for Windows 95/98 and Windows NT 3.51 • Monitor Client Library – an application programming interface that provides access to Adaptive Server performance data.

Product and Platforms	Description
Language modules version 12.0	Display system messages and datetime formats to help you localize your 32-bit or 16-bit applications. Default installation includes: • cp437, cp850, iso_1, mac, and roman_8 character sets • U.S. English language module • Additional language modules, such as Japanese and Chinese
jConnect 4.2	Provides a Java database connectivity (JDBC) driver that works with both Sun and Microsoft virtual machines (VM).jConnect 4.2 contains the latest enhancements to jConnect that can be used in a JDK 1.1 environment. The intent is to provide customers with the latest jConnect features for applications using a Java 1.1 VM or later. For more information jConnect for JDBC, see the jConnect product page at http://www.sybase.com/products/internet/jconnect/.
jConnect 5.2	Provides a Java database connectivity (JDBC) driver that works with both Sun and Microsoft virtual machines (VM). jConnect 4.2 provides support for new features available in Adaptive Server Enterprise 12.0 and support for JDBC 2.0 standard extensions and is JDBC 2.0 compliant. For more information jConnect for JDBC, see the jConnect product page at http://www.sybase.com/products/internet/jconnect/.
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 version 12.0	The graphical administration utility for Adaptive Server and PowerDynamo.
Sybase ODBC drivers version 3.5	PowerDesigner and PowerDynamo connect to Adaptive Server using the ODBC driver.
Power Dynamo version 1.2	A suite of tools for building and managing database-hosted Web sites with dynamic content.

Product and Platforms	Description
InfoMaker version 7.0.1	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 6.0.2	SQL Remote enables two-way replication between a database server and multiple portable computer databases through E-mail or dial-up connections.
PowerDesigner SQL Modeler version 6.1.4	A tool for data modeling, including database design, generation, maintenance, reverse engineering, and documentation for database architects.

Unloading PC-Client Products

To unload client products on Windows NT or Windows 95/98:

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 up memory and system resources.

4 Insert the CD containing the PC-client products into the CD-ROM drive.

5 The Studio Installer should start automatically. Alternatively you can start the Studio Installer by doing one of the following:

- Click the Windows Start button, choose Run, and enter:

  ```
  n:\setup.exe
  ```

  where *n* is your CD-ROM drive.

- Or, click the Start button, choose the Windows Explorer, select your CD-ROM drive, and then double-click *setup.exe*.

6 Enter the target directory.

7 Select the type of installation to be performed. Backup Server, Monitor Server, and XP Server are unloaded, by default, with Adaptive Server.

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

- *Full Install* – installs every component on the CD.
- *Customized Install* – allows you to select which components to install. Certain components will automatically be 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. Clicking this button allows you to select or deselect subcomponents.

8 Click Next to launch the Summary screen.

The Summary screen displays every component that will be installed by the Studio Installer. Disk space required for each selected component is also shown, along with the available disk space.

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

9 Click Next to continue the installation. The Studio Installer unloads the components from the CD and displays a progress indicator.

Warning! If you are prompted to overwrite any DLLs loaded in memory, select No.

If no component has been selected, the Studio Installer generates an error message and stops the installation. Click Back to select components, or Cancel to cancel the installation procedure.

Note Optionally, you may select Save from the Summary screen to save all the installation information into a *cmdfile* to proceed with the installation in a non-interactive, silent install. (See "Unloading Components in cmdfile" on page 33.)

If any components selected to be installed require a license, the Sybase Software Asset Management (SySAM) screen displays and prompts you for certification information at this time. If you install any components without the appropriate license information, only Adaptive Server 12.0 without licensed features is enabled.

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

To configure client network connections to Adaptive Server, see "Configuring Network Connections for Client Products" on page 104.

Note If you downloaded jConnect 4.2 or 5.2 from the PC-Client CD, set the JDBC_HOME and CLASSPATH environment variables and test the installation. You may also want to install stored procedures. See "setting jConnect Environment Variables, "Installing Stored Procedures for jConnect" and "jConnect 4.2 and 5.2" in Chapter 6, "Post-Installation Tasks".

Installing the Adaptive Server Plug-In for Sybase Central

This section describes how to install the Adaptive Server plug-in for Sybase Central. The plug-in is a graphical tool for managing the Adaptive Servers in your Sybase installation. Sybase Central can be installed on the same Windows NT, 95/98 computer as the server or on another computer.

Installing the Sybase Central Plug-In

To install the Sybase Central plug-in from your Sybase products CD, follow the instructions in "Unloading PC-Client Products" on page 94 and select the Sybase Central ASE Plug-in option in the Product Selection window.

Before you use the plug-in, restart your computer to implement changes made to the PATH and CLASSPATH environment variables during installation.

Installing the Required Microsoft DLL on Windows 95

After installing the Sybase Central plug-in on a computer that is running Windows 95, check the version of the *comctl32.dll* system file. This file provides common window and dialog graphical controls. If your system has an older version of this file, some Adaptive Server plug-in dialog boxes might not be displayed properly.

Checking the Current File Version

To check the file *comctl32.dll*:

1 Open the Windows NT Explorer to the *d:\windows\system* directory.

2 Locate the file *comctl32.dll*.

3 If the Modified date is earlier than 4/30/97, replace the file with a newer version from the Microsoft Web site.

You can check the modified date by opening the property sheet for the file or putting the right pane of the Windows NT Explorer in Details view and looking in the Modified column.

Replacing the File

To replace the file *comctl32.dll*:

1 Access the Microsoft Web site:

http://support.microsoft.com/support/kb/articles/Q165/4/87.asp

2 Get the file *com32upd.exe* from the Microsoft Web site. Follow the instructions for installation provided by Microsoft, but do not use the **-d** parameter.

3 After you install the update, restart your computer.

Installing a Java Runtime Environment

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

To install the JRE:

- Go to: *%SYBASE%\ASEP*.
- Double-click the file *jdk1_1_8-win.exe* and follow the installation instructions.
- Restart your computer after you install the JRE.

Installing the Entire 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).

To install the JDK:

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:

   *d:**install_path*\lib\CLASSES.ZIP

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

Enabling the Monitoring Functions

If you cannot run the monitoring windows in the Adaptive Server plug-in, check the following:

- Be sure the CLASSPATH variable is set, even though the JRE documentation may ask that you not set it. The CLASSPATH variable indicates where Java classes are located.

  For example, if the JRE is installed in the release directory %SYBASE%\%SYBASE_ASE\, the plug-in requires CLASSPATH to contain:

  %SYBASE%\%SYBASE_ASE\asep\monclass.zip;
  %SYBASE%\%SYBASE_ASE\3pclass.zip

  If you are using JDK 1.1.5 or later, the CLASSPATH must also include:

  %SYBASE%\install_path\lib\classes.zip

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

  Note The Adaptive Server plug-in installation process adds the following subdirectories to the PATH statement:
  %SYBASE%\Sybase\ASE\ASEP
  where *%SYBASE%\Sybase\ASE* is the Adaptive Server software location.

- Make sure that the value of the system variable CLASSPATH is not overridden by a user-level definition of the variable.

  If you install applications that set the user CLASSPATH variable, this can cause the system CLASSPATH, and therefore the monitor classes, to be ignored.

  To resolve this problem, set the user CLASSPATH to include the system CLASSPATH, as follows:

  ```
  User CLASSPATH=%CLASSPATH%;d:\new_path
  ```

  where *d:\new_path* is the path to the system CLASSPATH.

- If a Monitor Server's folder does not appear under an Adaptive Server Enterprise icon, see the Monitor Server tab on the Adaptive Server property sheet to associate the Monitor Server with the Adaptive Server.
- Make sure that Monitor Server is running.

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.

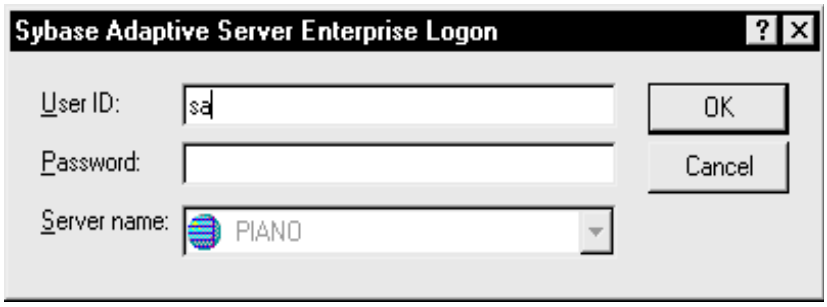
To test installation and network connections:

1 Restart the computer.

2 To start Sybase Central, select Start | Programs | Sybase Central from the Windows task bar.

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

4 Log in using the Server Config user ID. The installation program creates the default user ID “sa” without a password. You can change the password later through the Server Config utility.



Your User ID appears in parentheses next to the server name.

5 Click on each of the other Adaptive Server installations.

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. After Adaptive Server starts, the Sybase Adaptive Server Enterprise Logon dialog box displays.

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.

6 To disconnect from a server, choose Disconnect from the Tools menu.

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

7 Exit Sybase Central.

If Adaptive Server fails any of these tests, see Chapter 12, "Troubleshooting" Before retrying the installation, follow the instructions in Chapter 6, "Post-Installation Tasks".

Installing Open Client/C

Table 7-2 lists the Sybase products you can install separately to develop and run client applications for operating systems such as Windows NT, Windows 95, Windows 3.1, OS/2, and Macintosh.

Table 7-2: Additional Sybase products

Product	Installation instructions
Use Open Client/C for Windows NT and Windows 95/98 to develop and run applications to access Adaptive Server from a Windows client computer.	Run *setup.exe* from the Sybase products CD and select Open Client Runtime from the Select Installation Type window.

Installing Open Client Runtime for Windows 95/98 and NT

The Sybase PC-Client CD includes Open Client software for Windows NT and Windows 95/98. 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.

To install Open Client/C:

1 Log in to your Windows 95/98-based or Windows NT-based computer using an account with Windows administrator privileges.

2 Close any open applications or utilities to free up disk space and system resources.

3 Insert the CD into the CD-ROM drive.

4 The Studio Installer starts automatically. Alternatively, you may start the installation program by using one of the following methods:

   - For Windows 95/98 and Windows NT 4.0, click the Start button, choose Run, and type:

     ```
     d:\setup.exe
     ```

     where *d:* is your CD-ROM drive.

   - Alternatively, start the installation program by choosing the Windows NT Explorer from the Start button, selecting the CD-ROM drive, and then double-clicking *setup.exe*.

5 Enter the target directory.

6 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 which components to install. Certain components will automatically be 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. Clicking this button allows you to select or deselect subcomponents.

7 Choose Open Client Runtime as the installation, and click Next in the Select Installation Type window. The Select Sybase Installation Directory window displays.

8 Click Next to launch the Summary screen.

The Summary screen displays every component that will be installed by the Studio Installer. Disk space required for each selected component is also shown, along with the available disk space.

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

9 Click Next to continue the installation. The Studio Installer unloads the components from the CD and displays a progress indicator.

Warning! If you are prompted to overwrite any DLLs loaded in memory, select No.

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

Note Optionally, you may select Save from the Summary screen to save all the installation information into a *cmdfile* to proceed with the installation in a non-interactive, silent install. (See "Unloading Components in cmdfile" on page 33.)

If any components selected to be installed require a license, the Sybase Software Asset Management (SySAM) screen displays and prompts you for certification information at this time. If you install any components without the appropriate license information, only Adaptive Server 12.0 without licensed features is enabled.

11 Use the Windows Notepad to check the installation log file if you encounter problems. The file is automatically generated in:

%SYBASE%\installer.log.

Installing InfoMaker

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

To install InfoMaker:

1 Use the Studio Installer to unload the software from the distribution media as described in Chapter 3, “Unloading Server Products from Distribution Media”.

2 Run the InfoMaker setup utility located in:

%SYBASE%\info-7_0_1\install\setup.exe

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

4 When prompted to reboot 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.

Configuring Network Connections for Client Products

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

Warning! The *libtcl.cfg* file, located in *%SYBASE%\ini* will be overwritten, if it already exists.

Network connections between clients and servers are defined in the *sql.ini* file on the PC. Sybase client products use *sql.ini* to find the network addresses of available servers.

Before any client product can access a Sybase server, each server you want to connect to must be included in *sql.ini* on the client computer.

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

Adding a Server to the *sql.ini* File

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

1 Using the Microsoft Windows task bar, click 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, and 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, SPX, or VINESIP.

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

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.

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 a Solaris machine, the host number is given in a hexidecimal. Use one of these techniques to decode it:
  - Run the **dsedit** program from the */bin* directory below SYBASE installation directory. When the server name list appears, double-click the server name. The next screen shows the port number in decimal. Exit without making any changes.
  - Check the mast line from the interfaces file. The port number appears in hexidecimal in the four digits following the "\x0002". For example:

```
testserver
      master tli tcp /dev/tcp \x0002100482d63ce70000000000000000
       query tli tcp /dev/tcp \x0002100482d63ce70000000000000000
```

In this example, the port number is 1004 (hexadecimal). When 1004 (hexadecimal) is converted to decimal, the port number is 4100. The port number must be between the decimal value of 1025 and 65535.

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

```
testserver
query=TCP, machine_name, 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 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:

- *Adaptive Server Enterprise Monitor Server User's Guide*
- *Adaptive Server Enterprise Monitor Historical Server User's Guide*

For information about using **dsedit** on the server machine or for more information on the components of an interfaces file entry, see Chapter 13, "Configuring the Operating System".

Enabling TCP Connections

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

To enable TCP connections:

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

2 Click the NetLibrary tab.

3 Select Windows NT or 98/95 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*).

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.

See *Configuring Adaptive Server Enterprise for Windows NT* for more information.

What's Next?

You have completed installing Adaptive Server client plug-ins.

- To find out more information about setting up network communications, see *Configuring Adaptive Server Enterprise for Windows NT*.
- To upgrade Adaptive Server from a previous release, see Chapter 9, "Upgrading Sybase Servers".
- To add optional functionality, see Chapter 10, "Adding Optional Functionality to Adaptive Server".
- After you have installed the Sybase products on your system, see their accompanying documentation for configuration and administration issues.

CHAPTER 8

Customizing Localization for Adaptive Server

This chapter provides information about Sybase localization support for international installations. It also includes information for reconfiguring localization.

Topics covered are:

Name	Page
Overview of Localization Support	109
Character Sets	111
Character-Set Conversion	116
Sort Orders	120
Language Modules	126
Localization	127
Changing the Localization Configuration	131

This chapter provides only the information that you need to know to configure languages, character sets, and sort order. For more information, see the *System Administration Guide*.

Overview of Localization Support

Localization is the process of setting up an application to run in a particular language or country environment, including translated system messages and correct formats for date, time, and currency. Adaptive Server supports localization for international customers and for customers with heterogeneous environments.

This support includes:

- Data processing support – Adaptive Server comes with character-set and sort-order definition files it uses to process the characters used in different languages.

Sybase provides support for the major languages in:

- Western Europe
- Eastern Europe
- Middle East
- Latin America
- Asia

- Translated system messages – Adaptive Server 12.0 includes language modules for:
  - Brazilian Portuguese
  - Chinese (Simplified)
  - French
  - German
  - Japanese
  - Korean
  - Spanish
- Translated documentation – Translated documentation is available in:
  - Chinese (Simplified)
  - French
  - German
  - Japanese
  - Korean
  - Spanish

Language Modules

Adaptive Server stores its localized software messages in separate language modules, each of which includes the following:

- Character sets
- Sort orders
- Messages

When you install a language module, the installation program loads the messages, character-set, and sort-order files that support the new language in the correct locations.

Note The Adaptive Server language modules do not contain error messages for standalone utilities such as **bcp**, **isq**, and **defncopy**. These error messages are contained in the Client-Library™ language modules. For information on running these utilities with a language other than U.S. English or a character set other than ISO 8859-1, see *Utility Programs for Windows and Windows NT*.

When you install Adaptive Server and Backup Server, system messages in English are installed by default.

Character Sets

This section contains information about the default and supported character sets.

Default Character Sets for Servers

The default character set is the character set in which data is encoded and stored on the Adaptive Server databases.

By default, when Adaptive Server and Backup Server are installed on SUN Solaris systems, the installation installs the character set files for ISO 8859-1 which supports the Western European languages.

Changing the Default Character Set for Servers

You can select any character set as the default on Adaptive Server, including character sets that are not the platform default character sets. Keep the following guidelines in mind when selecting a new default character set:

- To avoid conversion errors or overhead, determine the default character set based on the character set used by your clients.

  For example, if most of your clients use ISO 8859-1, you can minimize the amount of data conversion that has to occur by specifying ISO 8859-1.

- If your server is operating in a heterogeneous environment, choose a character set that will work with all the character sets needed. Often, this is Unicode (UTF-8).

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*.

Supported Character Sets

The following language, scripts and character sets are supported by Adaptive Server version 12.0:

- Arabic – See Table 8-1.
- Baltic – See Table 8-2.
- Chinese, Simplified – See Table 8-3.
- Chinese, Traditional – See Table 8-4
- Cyrillic – See Table 8-5.
- Eastern European – See Table 8-6.
- Greek – See Table 8-7.
- Hebrew – See Table 8-8.
- Japanese – See Table 8-9.
- Korean – See Table 8-10.
- Thai – See Table 8-11.
- Turkish – See Table 8-12.
- Unicode – See Table 8-13.
- Vietnamese – See Table 8-14.
- Western European – See Table 8-15.

The tables define each character set and indicate information on whether it requires Unilib ™ conversion (Unilib Required column).

- Checkmark (x) – the character set requires Unilib conversion.
- No checkmark – the character set may use either the Unilib conversion or the built-in conversion.

For more information see “Character-Set Conversion” on page 116.

Table 8-1: Arabic character sets

Character Set	Unilib Required	Description
cp864	X	PC Arabic
cp1256	X	Microsoft Windows Arabic
iso88596	X	ISO 8859-6 Latin/Arabic

Table 8-2: Baltic character sets

Character Set	Unilib Required	Description
cp1257	X	Microsoft Windows Baltic

Table 8-3: Simplified Chinese character sets

Character Set	Unilib Required	Description
eucgb	X	EUC GB encoding = Simplified Chinese character sets
cp936	X	Microsoft Simplified Chinese character sets

Table 8-4: Traditional Chinese character set

Character Set	Unilib Required	Description
cp950	X	PC (Microsoft) Traditional Chinese
euccns	X	EUC CNS encoding = Traditional Chinese with extensions
big5	X	Big 5 Traditional Chinese

Table 8-5: Cyrillic character sets

Character Set	Unilib Required	Description
cp855		IBM PC Cyrillic
cp866		PC Russian
cp1251		Microsoft Windows 3.1 Cyrillic
iso88595		ISO 8859-5 Latin/Cyrillic
koi8		KOI-8 Cyrillic
mac_cyr		Macintosh Cyrillic

Table 8-6: Eastern European character sets

Character Set	Unilib Required	Description
cp852		PC Eastern Europe
cp1250		Microsoft Windows 3.1 Eastern European
iso88592		ISO 8859-2 Latin-2
mac_ee		Macintosh Eastern European

Table 8-7: Greek character sets

Character Set	Unilib Required	Description
cp869		IBM PC Greek
cp1253		MS Windows Greek
greek8		HP GREEK8
iso88597		ISO 8859-7 Latin/Greek
macgrk2		Macintosh Greek

Table 8-8: Hebrew character sets

Character Set	Unilib Required	Description
cp1255	X	Microsoft Windows Hebrew
iso88598	X	ISO 8859-8 Hebrew

Table 8-9: Japanese character sets

Character Set	Unilib Required	Description
cp932	X	IBM J-DBCS:CP897 + CP301 (Shift-JIS)
deckanji		Digital UNIX JIS encoding
eucjis		EUC-JIS encoding
sjis		Shift-JIS (no extensions)

Table 8-10: Korean character sets

Character Set	Unilib Required	Description
eucksc	X	EUC KSC Korean encoding = CP949

Table 8-11: Thai client character sets

Character Set	Unilib Required	Description
tis620	X	TIS-620 Thai standard
cp874	X	Microsoft Windows Thai

Table 8-12: Turkish character sets

Character Set	Unilib Required	Description
cp857		IBM PC Turkish
cp1254		Microsoft Windows Turkish
iso88599		ISO 8859-9 Latin-5 Turkish
macturk		Macintosh Turkish
turkish8		HP TURKISH8

Table 8-13: Unicode character set

Character Set	Unilib Required	Description
utf8	X	Unicode UTF-8 encoding

Table 8-14: Vietnamese character set

Character Set	Unilib Required	Description
cp1258	X	Microsoft Windows Vietnamese

Table 8-15: Western European character set

Character Set	Unilib Required	Description
ascii8	X	US ASCII, with 8-bit data, ISO 646
cp437		IBM CP437 - U.S. code set
cp850		IBM CP850 - European code set
cp860	X	PC Portuguese
cp863	X	IBM PC Canadian French code page
cp1252	X	Microsoft Windows US (ANSI)
iso_1		ISO 8859-1 Latin-1
mac		Standard Macintosh coding
roman8		HP ROMAN8
iso 885915	X	ISO 8859-15 Latin-1 with Euro support

Character-Set Conversion

Backup Server passes messages to Adaptive Server in the client's language and in the Adaptive Server character set. Adaptive Server then converts the messages and issues them in the client's language and character set. Keep the following requirements in mind when selecting a character set:

- In a heterogeneous environment, Adaptive Server and Backup Server may need to communicate with clients running on different platforms and using different character sets. To maintain data integrity, the server converts the code between the character sets.
- To use the built-in conversion, you need to install the character-set definition files on the server for all the character sets being used by your clients. Built-in conversion support is available for many character sets.
- Unilib conversion support is available for all character sets supported by Sybase. To enable Unilib conversion, you must use **sp_configure** and turn **enable unicode conversions** on. For more information see the *System Administration Guide*.

If either Adaptive Server or Backup Server does not support a client's language or character set, that server issues a warning message. Errors also occur when the Backup Server character set is not compatible with the Adaptive Server character set.

Character-set conversion is supported only between character sets for the same language or between character sets in the same language group.

For example, automatic character-set conversion is supported between the character sets for the Western European languages: ASCII 8, CP 437, CP 850, CP 860, CP 863, CP 1252, ISO 8859-1, ISO 8859-15, Macintosh Roman and ROMAN8. Similarly, conversion is supported between the character sets for Japanese: CP 932, EUC-JIS, Shift-JIS, and DEC-Kanji.

However, code conversion is not supported between any of the Western European language character sets and the Japanese character sets. For more information about supported conversions, see the *System Administration Guide*.

Conversions Between Server and Client

If Adaptive Server does not support the client's language or character set, the client can connect with the server, but no character conversions will occur.

When a localized client application connects to Adaptive Server, the server checks to see if it supports the client's language and character set.

- If Adaptive Server supports the language, it automatically performs all character-set conversions and displays its messages in the client's language and character set.
- If Adaptive Server does not support the language, it uses the user's default language or Adaptive Server's default language.
- If Adaptive Server does not support the character set, it issues a warning to the client, turns conversion off, and sets the language to U.S. English.

Limitations of Hankaku Katakana Conversion

Note This is not a problem if the server is set to use Unicode conversions and the client is using Open Client 11.1 or later.

Hankaku Katakana (half-width phonetic Japanese) conversions have the following limitations and rules:

- Hankaku Katakana conversion is enabled only if Adaptive Server is using EUC-JIS, and clients are using Shift-JIS.
- Hankaku Katakana character-set conversion between Shift-JIS and EUC-JIS by default is not enabled in Adaptive Server.

  You can, however, enable Hankaku Katakana character-set conversion by turning on trace flag 2402 when starting Adaptive Server. Include "-T2402" on the command line in the script used to start Adaptive Server.
- After Hankaku Katakana conversion is enabled, you cannot change the conversion on a per-client basis.

  To turn off the conversion, either remove "-T2402" from the script or use a different script with the trace flag disabled. Then restart Adaptive Server.
- If conversion is disabled for Hankaku Katakana, each unconverted Hankaku character is replaced with a question mark (?).
- The client application using Shift-JIS must recognize different character sizes when defining a table. Data that is longer than column length is truncated. No warning message appears.
- In an Adaptive Server using EUC-JIS, characters are stored in the EUC-JIS sort order. Zenkaku characters come after Hankaku characters in the EUC-JIS character set. Therefore, Adaptive Server sorts Hankaku first, followed by Zenkaku.
- If the column contains Hankaku Katakana, the Adaptive Server using EUC-JIS sends the client application using Shift-JIS converted characters to pad the length with NULLs.
- Hankaku Katakana can be used in logical expressions, for example:

  ```
  1> select * from table
  2> where column=Hankaku_Katakana_here
  3> go
  ```
- The Transact-SQL search functions treat Hankaku Katakana and Zenkaku Katakana as different sets of characters. Katakana can be either half-width (Hankaku) or full-width (Zenkaku) in both EUC-JIS and Shift-JIS.

  For example, if the data in a table is stored in Hankaku Katakana, the following SQL statement returns no rows, even if both Hankaku and Zenkaku data have the same meaning:

  ```
  1> select * from table
  ```

```
2> where column=Zenkaku_Katakana_here
3> go
```

- Hankaku Katakana cannot be used for procedure names or parameter names because the internal memory allocated for these names may be less than the length required by EUC-JIS.
- You can use **bcp** from a client application using Shift-JIS to an Adaptive Server using EUC-JIS; however, you cannot run **bcp** from Shift-JIS client applications containing Hankaku Katakana characters.

Hankaku Katakana Conversion and Datatypes

The following limitations apply to datatypes in Hankaku Katakana conversion:

- Hankaku Katakana can be used only in *char*, *varchar*, *nchar*, *nvarchar*, and *text* datatypes.
- The *char*, *varchar*, *nchar*, *nvarchar*, and *text* datatypes are stored in an Adaptive Server using EUC-JIS and can be used for a Shift-JIS client application.

  However, because one Shift-JIS Hankaku character uses 1 byte of storage and one EUC-JIS Hankaku character uses 2 bytes of storage, a client application using Shift-JIS can insert a maximum of 127 Hankaku Katakana characters, rather than the 255 characters of Adaptive Server using Shift-JIS. Inserting more than 127 characters during conversion to EUC-JIS generates an error message.
- Sybase recommends that you not use Hankaku Katakana for the *sysname* datatype, such as database names, table names, column names, and so on, because of the conversion length changes.
- If you plan to convert with Shift-JIS, do not use Hankaku Katakana in *text* datatypes that are larger than one data page, typically 2K.

  If the text column contains Hankaku Katakana, and the character spans two pages in the database, the partial byte in the first page is not converted. A warning message advises that the conversion did not succeed.

  There is no problem when converting EUC-JIS to EUC-JIS or Shift-JIS to Shift-JIS.

Sort Orders

Each character set comes with one or more sort orders (collating sequences), which are located in the sort-order definition files (*.srt* files). These files accompany the character-set definition files and can be found in the same directory.

You can select a sort order for your data according to the needs at your site. However, keep in mind that the server can support only one sort order at a time, so be sure to select a sort order that will work for all of your clients.

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 may cause incorrect behavior. To change the character set or sort order after you have added data, see the *System Administration Guide.*

Available Sort Orders

The sort order determines the collating sequence Adaptive Server uses to order, compare, and index character data. Each character set comes with one or more sort orders.

Sort orders are located in sort order definition files (*.srt* files) that accompany your character-set definition files.

Note Available sort orders vary according to the character set installed on Adaptive Server.

You can see the available sort orders for your character set by looking in the *.srt* file for your language. Sort orders are stored in the following path:

$SYBASE/charsets/<charset_name>/.srt*

For more information about localization files, see “Localization Directories” on page 127.

Table 8-16 describes the sort orders that you can specify at installation time or at a later time using the **sqlloc** utility.

Table 8-16: Sort orders available in Adaptive Server

Sort Order Name	Description
Binary order	Sorts all data according to numeric byte values for that character set. Binary order sorts all ASCII uppercase letters before lowercase letters. Accented or ideographic (multibyte) characters sort in their respective standards order, which may be arbitrary. All character sets have binary order as the default. If binary order does not meet your needs, you can specify one of the other sort orders either at installation or at a later time by, using the sqlloc utility.
Dictionary order case sensitive, accent sensitive	Case sensitive. Sorts each uppercase letter before its lowercase counterpart, including accented characters. Recognizes the various accented forms of a letter and sorts them after the associated unaccented letter.
Dictionary order case insensitive, accent sensitive	Case-insensitive dictionary sort order. Uppercase letters are equivalent to their lowercase counterparts and are intermingled in sorting results.
Dictionary order case insensitive accent insensitive	Case-insensitive dictionary sort order. Diacritical marks are ignored.
Dictionary order case insensitive with preference	Case-insensitive dictionary sort order, with case preference for collating purposes. A word written with uppercase letters is equivalent to the same word written with lowercase letters. Uppercase and lowercase letters are distinguished only when you use an order by clause. The order by clause sorts uppercase letters before it sorts lowercase. **Note** Do not select this sort order unless your installation requires that uppercase letters be sorted before lowercase letters in otherwise equivalent strings for order by clauses. Using this sort order may reduce performance in large tables when the columns specified in an order by clause match the key of the table's clustered index.
Alternate dictionary order case sensitive	Case-sensitive alternate dictionary sort order with lowercase variants sorted before uppercase. Use with several of the Western European languages.
Alternate dictionary order case insensitive accent insensitive	Case-insensitive and accent-insensitive alternate dictionary sort order. Use with several of the Western European languages.

Sort Order Name	Description
Alternate dictionary order case insensitive uppercase preference	Case-insensitive alternate dictionary sort order with uppercase preference. Use with several of the Western European languages.

To see the sort orders that are available, use **sqlloc** to display the sort orders for the character sets you plan to use.

Gathering Information About Collation

The following functions enable you to perform collation operations in your applications:

- **sortkey** – generates binary values that you can use to order query results based on collation behavior.
- **compare** – compares two character strings based on alternate collation rules.

These functions can be nested and used anywhere that an expression is allowed, including stored procedures and triggers.

Ordering Query Results Based on Collation Behavior

The **sortkey** function generates values that can be used to order results based on collation behavior. This allows you to work with character collation behaviors beyond the default set of Latin-character-based dictionary sort orders and case or accent sensitivity. The return value is a *varbinary* datatype value that contains coded collation information for the input string that is retained from the **sortkey** function.

For example, you can store the values returned by **sortkey** in a column with the source character string. When you want to retrieve the character data in the desired order, the **select** statement only needs to include an **order by** clause on the columns that contain the results of running **sortkey**.

The **sortkey** function guarantees that the values it returns for a given set of collation criteria work for the binary comparisons that are performed on varbinary datatypes.

Syntax for sortkey

The syntax is:

sortkey (*char_expression* [, {*collation_name* | *collation_ID*}])

where:

- *char_expression* is one of the following:
  - Character type (*char*, *varchar*, *nchar*, or *nvarchar*)
  - Character variable, or
  - Constant character expression, enclosed in single or double quotation marks
- *collation_name* is a quoted string or a character variable that specifies the collation to use.
- *collation_ID* is an integer constant or a variable that specifies the collation to use.

There are two types of collation tables, built-in and external. You can use either the collation name or the collation ID to specify a built-in table. You must use the collation name to specify external tables. Table 8-17 lists the valid values for *collation_name* and *collation_ID*.

Table 8-17: Collation Names and IDs

Description	Collation Name	Collation ID
Binary sort	binary	50
Default Unicode multilingual	default	0
CP 850 Alternative: no accent	altnoacc	39
CP 850 Alternative: lower case first	altdict	45
CP 850 Alternative: no case preference	altnocsp	46
CP 850 Scandinavian dictionary	scandict	47
CP 850 Scandinavian no case preference	scannocp	48
GB Pinyin	gbpinyin	n/a
Latin-1 English, French, German dictionary	dict	51
Latin-1 English, French, German no case	nocase	52
Latin-1 English, French, German no case preference	nocasep	53
Latin-1 English, French, German no accent	noaccent	54
Latin-1 Spanish dictionary	espdict	55
Latin-1 Spanish no case	espnocs	56
Latin-1 Spanish no accent	espnoac	57
ISO 8859-5 Cyrillic dictionary	cyrdict	n/a

Description	Collation Name	Collation ID
ISO 8859-5 Russian dictionary	rusdict	n/a
ISO 8859-9 Turkish dictionary	turdict	n/a
Shift-JIS binary order	sjisbin	n/a
Thai dictionary	thaidict	1

Rules for Using sortkey

The following rules apply to using the **sortkey** function:

- *char_expression* must be composed of characters that are encoded in the server's default character set.
- *char_expression* can be an empty string. If it is an empty string:
  - **sortkey** returns a zero-length varbinary value, and
  - Adaptive Server stores a blank for the empty string.

  An empty string has a different collation value than an NULL string from a database column.
- If *char_expression* is NULL, **sortkey** returns a NULL value.
- If you do not specify a value for *collation_name* or *collation_ID*, **sortkey** assumes binary collation.

Note **sortkey** can generate up to 6 bytes of collation information for each input character. Therefore, the result from using **sortkey** may exceed the 255-byte length limit of the varbinary datatype. If this happens, the result is truncated to fit. Truncation removes result bytes for each input character until the result string is less than 255 bytes. If this occurs, a warning message is issued, but the query or transaction that contained the **sortkey** function continues to work.

Comparing Strings Based on Alternate Collation Rules

The **compare** function allows you to directly compare two character strings based on alternate collation rules.

Syntax for *compare*

The syntax is:

```
compare (char_expression1, char_expression2
[, {collation_name | collation_ID}])
```

where:

- *char_expression1* is the character expression you want to compare to *char_expression2*.

  char_expression2 is the character expression against which you want to compare *char_expression1*.

  char_expression1 and *char_expression2* can be one of:

  - Character type (*char*, *varchar*, *nchar*, or *nvarchar*)
  - Character variable
  - Constant character expression, enclosed in single or double quotation marks
- *collation_name* can be a quoted string or a character variable that specifies the collation to use.
- *collation_ID* is an integer constant or a variable that specifies the collation to use.

The **compare** function returns the following values, based on the collation rules that you chose:

- 1 – indicates that *char_expression1* is greater than *char_expression2*.
- 0 – indicates that *char_expression1* is equal to *char_expression2*.
- -1 – indicates that *char_expression1* is less than *char_expression2*.

Rules for Using *compare*

The following rules apply to using the **compare** function:

- Both *char_expression1* and *char_expression2* must be characters that are encoded in the server's default character set.
- Either *char_expression1* or *char_expression2*, or both, can be empty strings:
  - If *char_expression2* is empty, the function returns 1.
  - If both strings are empty, then they are equal, and the function returns a 0 value.
  - If *char_expression1* is empty, the function returns a -1.

The **compare** function does not equate empty strings and strings containing only spaces, as Adaptive Server does. **compare** uses the **sortkey** function to generate collation keys for comparison. Therefore, a truly empty string, a string with one space, or a string with two spaces do not compare equally.

- If either *char_expression1* or *char_expression2* is NULL, then the result will be NULL.
- If you do not specify a value for *collation_name*, **compare** assumes binary collation.
- If you do not specify a value for *collation_ID*, **compare** assumes binary collation.

Language Modules

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

When you install a new language module, installation automatically loads the language, character set and sort order files into the Sybase installation directory to support the new language. For information about directories, see "Localization Directories" on page 127.

Installing a New Language Module

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

To install a new language module:

1 Load the language module software from the distribution media. You must load this software into the same directory in which you loaded Adaptive Server.

2 Reconfigure the language and, if necessary, the character set and sort order for Adaptive Server. For instructions, see "Changing the Localization Configuration" on page 131.

Message Languages

For messages, U.S. English is installed as the default language in Adaptive Server. The following rules apply to language modules:

- During Adaptive Server installation or reconfiguration, you can specify a default language other than U.S. English. However, you must have installed the language module for the language you specify.
- If your clients require Adaptive Server messages in a language other than U.S. English, you must load the language module for those languages. Then, you can configure Adaptive Server to the language used by your clients.
- If Adaptive Server does not support messages in a client's language, these clients receive messages in the Server default language.

  For example, if your client's language is Latin, the Spanish language module is installed, and Spanish is specified as the Adaptive Server default language, the client receives messages in Spanish.

Localization

By default, the Adaptive Server and Backup Server configurations use the English locale settings, which include:

- Character-set definition files for Western European character sets
- Sort-order definition files for Western European character sets
- U.S. English system message files

During the installation process or through reconfiguration, you can specify a different language, character set, and sort order.

Localization Directories

Sybase localization configuration involves the following directories:

- *locales*
- *charsets*

The table below illustrates the structure of the localization files. It does not show a complete list of all the files.

%SYBASE% or *$SYBASE/*	*charsets*	*charset_name*	**.srt* files
		charset_name...	*charset.loc*
		unicode	**.uct* files
	locales	*language_name*	charset_name
		language_name...	*charset_name...*
		locales.dat	
		message	*language_name*
			language_name...

About the Directory

The *$SYBASE/locales* directory contains a subdirectory for each available language. Each language subdirectory contains a subdirectory for each character set available with that language.

- The *.loc* files in these subdirectories enable Adaptive Server or Backup Server to report errors in a specific language, encoded in a specific character set.

  There are a variety of *.loc* files in each subdirectory. Most of these files contain translated error messages for a specific product or utility.
- The *common.loc* file in each subdirectory contains localized information, such as local date, time, and currency formatting, that is used by all products.
- The *locales.dat* file contains entries that associate platform-specific locale names with Sybase language and character-set combinations.

About the *charsets* Directory

The files in *$SYBASE/charsets/charset_name* contain information related to each particular character set, such as the definition of the character set and any sort orders available for that character set.

About the *locales.dat* File

You can edit the *locales.dat* file to:

- Change the default language or character set for a platform, or
- Add new associations between platform locale names and Sybase language and character-set names.

Format of *locales.dat* File Entries

Each entry in the *locales.dat* file links a platform-specific locale definition to a Sybase language and character set combination. Each entry has the following format:

```
locale = platform_locale, syb_language, syb_charset
```

where:

- *platform_locale* is the platform-specific keyword for a locale. For acceptable values, see your operating system documentation.

  When the locale being defined is the default for the site, *platform_locale* is “default”.

- *syb_language* is the name of the language directory to be used from within *$SYBASE/locales/language_name*.
- *syb_charset* is the character-set name that determines the character-set conversion method and identifies the directory location of the message files for clients from within *$SYBASE/locales/charset_name*.

For example, the following entry specifies that the default locale uses us_english for the language and iso_1 for the character set:

```
locale = default, us_english, iso_1
```

How Client Applications Use *locales.dat*

Client applications use the *locales.dat* file to identify the language and character set to use. The connection process follows these steps:

1 When a client application starts, it checks the operating system locale setting and then checks the *locales.date* file to see if that setting is appropriate for Adaptive Server. For example, a locale entry for French can look like the following:

```
locale = fr_FR, french, iso_1
```

2 When the client connects to Adaptive Server, the language and character-set information is passed to Adaptive Server in the login record.

3 Adaptive Server then uses:

- The character-set information, for example, iso_1, to identify the client's character set and verify whether it can convert character data to this character set
- The language (in the preceding example, French) and character-set information to see if it has messages in the client's language

Note Adaptive Server software comes with some locale entries already defined in the *locales.dat* file. If these entries do not meet your needs, you can either modify them or add new locale entries.

Editing the *locales.dat* File

Before beginning the edit, make a copy of the original file, in case you have problems with the resulting edited version.

To edit the *locales.dat* file:

1 Open the *locales.dat* file copy in a text editor such as Notepad.

2 Find the section for Sun Solaris, which is enclosed in brackets *[sun_svr4]*.

3 Make sure the section contains an entry for the language (*syb_language*) and character set (*syb_charset*) combination that you want to use.

- If an entry does not exist, continue with step 4.
- If an entry does exist, continue with step 5.

Note The value for *platform_locale* must match the value required by your operating system. If the locales definitions in your system configuration files do not match the Sybase locale definitions, your applications will not run properly.

For example, if you want your Open Client messages to appear in French, and Adaptive Server is using the ROMAN8 character set, you would check the *locales.dat* entries for your platform and look for the following entry:

```
locale = fr_FR, french, roman8
```

4 Add the required entry or modify an existing entry.

5 Save the changes, if any, and exit the text editor.

Changing the Localization Configuration

By default, the Adaptive Server and Backup Server configurations uses the English locale settings localization, which include:

- Character-set definition files for Western European character sets
- Sort order definition files for Western European character sets
- us_english system message files

During the installation process and through reconfiguration, you can specify a different language, character set, and sort order.

For Adaptive Server

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.

Note If you want to install more than one language on Adaptive Server, and the *master* database is not large enough to manage more than one language, the transaction log may become too full. You can expand the *master* database only on the master device. For more information, see the *System Administration Guide*.

---

1 To configure localization for Adaptive Server on the server, start **sqlloc**:

```
$SYBASE/$SYBASE_ASE/bin/sqlloc
```

2 Select Localize an exiting server.

3 From the Adaptive Server selection window, select the server.

4 Supply the user name and password. The user must have “sa” privileges.

5 Next, supply the following:

- Select the default language

- Select the default character set
- Select the default sort order

6 Select any other languages you want to install. You may only select languages that are supported by the default character set.

The Add and Remove Languages window lists all Sybase supported languages.

7 The Localization Summary window summarizes the configuration options you selected. Click OK to confirm your selections.

The Status Output window notifies you upon completing the installation.

For Backup Server

When you select the Backup Server to configure, Server Config displays the Configure Backup Server dialog box.

At the time you install the Backup Server, specify the language and default character set specified for Adaptive Server.

Configuring Adaptive Server for Other Character Sets

To configure Adaptive Server with the character set and sort order for your language, complete the following steps. Your system messages appear in the default language, English.

1 Use the **charset** utility to load the default character set and sort order.

To use the **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
```

Replace *sort_order_file* with the name of the sort order file. See Table 8-18 on page 133. Replace *character_set* with the Sybase name for your character set. See Table 8-19 on page 136.

2 Use the **charset** utility to load any additional character sets. See“charset Utility” on page 137 for more about this utility.

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

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

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

4 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
```

Replace *sort_order_id* with the ID for your sort order. See Table 8-18 on page 133. Replace *character_set* with the Sybase name for your character set. See Table 8-19 on page 136.

5 Shut down the server to start the reconfiguration process.

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

7 The server boots, rebuilds all the system indexes, then shuts down. Reboot a second time to bring the server up in a stable state.

Sort Orders

Table 8-18 describes the available sort orders. If your language does not appear, then there is no language-specific sort order for your language—use a binary sort order.

Table 8-18: Available sort orders

Language or Script	Sort orders	File name	ID
All languages	Binary order	*binary.srt*	50
Cyrillic	Dictionary order, case sensitive, accent sensitive	*cyrdict.srt*	63
	Dictionary order, case sensitive, accent sensitive	*cyrnocs.srt*	64

Language or Script	Sort orders	File name	ID
English French German (These sort orders work with all Western European character sets.)	Dictionary order, case sensitive, accent sensitive	*dictiona.srt*	51
	Dictionary order, case insensitive, accent sensitive	*nocase.srt*	52
	Dictionary order, case sensitive, accent sensitive, with preference	*nocasepr.srt*	53
	Dictionary order, case insensitive, accent insensitive	*noaccent.srt*	54
English French German (These sort orders work only with CP 850.)	Alternate dictionary order, case sensitive	*altdict.srt*	45
	Alternate dictionary order, case sensitive, accent insensitive	*altnoacc.srt*	39
	Alternate dictionary order, case sensitive, with preference	*altnocsp.srt*	46
Greek (This sort order works only with ISO 8859-7.)	Dictionary order, case sensitive, accent sensitive	*elldict.srt*	65
Hungarian (These sort orders work only with ISO 8859-2.)	Dictionary order, case sensitive, accent sensitive	*hundict.srt*	69
	Dictionary order, case insensitive, accent sensitive	*hunnoac.srt*	70
	Dictionary order, case insensitive, accent insensitive	*hunnocs.srt*	71
Russian (This sort order works with all Cyrillic character sets except for CP 855.)	Dictionary order, case sensitive, accent sensitive	*rusdict.srt*	58
	Dictionary order, case insensitive, accent sensitive	*rusnocs.srt*	59
Scandinavian (These sort orders work only with CP 850.)	Dictionary order, case sensitive, accent sensitive	*scandict.srt*	47
	Dictionary order, case insensitive, with preference	*scannocp.srt*	48

Language or Script	Sort orders	File name	ID
Spanish	Dictionary order, case sensitive, accent sensitive	*espdict.srt*	55
	Dictionary order, case insensitive, accent sensitive	*espnocs.srt*	56
	Dictionary order, case insensitive, accent insensitive	*espnoac.srt*	57
Thai	Dictionary order	*dictionary.srt*	51
Turkish (These sort orders work only with ISO 8859-9.)	Dictionary order, case sensitive, accent sensitive	*turdict.srt*	72
	Dictionary order, case insensitive, accent insensitive	*turnoac.srt*	73
	Dictionary order, case insensitive, accent sensitive	*turnocs.srt*	74

Character sets

Table 8-19 lists the supported character sets and their Sybase name.

Table 8-19: Sybase character-set names

Character sets	Sybase name
ASCII 8	acsii_8
Big 5	big5
CP 437	cp437
CP 850	cp850
CP 852	cp852
CP 855	cp855
CP 857	cp857
CP 860	cp860
CP 863	cp863
CP 864	cp864
CP 866	cp866
CP 869	cp869
CP 874	cp874
CP 932	cp932
CP 936	cp936
CP 950	cp950
CP 1250	cp1250
CP 1251	cp1251
CP 1252	cp1252
CP 1253	cp1253
CP 1254	cp1254
CP 1255	cp1255
CP 1256	cp1256
CP 1257	cp1257
CP 1258	cp1258
DEC Kanji	deckanji
EUC-CNS	euccns
EUC-GB	eucgb
EUC-JIS	eucjis
EUC-KSC	eucksc
GREEK8	greek8
ISO 8859-1	iso_1
ISO 8859-2	iso88592

Character sets	Sybase name
ISO 8859-5	iso88595
ISO 8859-6	iso88596
ISO 8859-7	iso88597
ISO 8859-8	iso88598
ISO 8859-9	iso88599
ISO 8859-15	iso885915
Koi8	koi8
Macintosh Cyrillic	mac_cyr
Macintosh Central European	mac_ee
Macintosh Greek	macgrk2
Macintosh Roman	mac
Macintosh Turkish	macturk
ROMAN8	roman8
Shift-JIS	sjis
TIS 620	tis620
TURKISH8	turkish8
UTF-8	utf8

charset Utility

Use the **charset** utility to load character sets and sort orders into Adaptive Server. If you are using **charset** to load the default character set and sort order, this should be done only at the time of installation.

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

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 ]
```

Table 8-20: Keywords and options for charsets

Keywords and Options	Description
-U	If you are not already logged in to your operating system as “sa”, you must specify “-Usa” or “/username = sa” in the command line.
-P	Specifies the “sa” password on the command line. If not specified, the user is prompted for the “sa” password.
-S	Specifies the name of the server. If not specified, charset uses the DSQUERY environment variable to identify the server name. If there is no DSQUERY environment variable, charset attempts to connect to a server named “SYBASE.”
-I	Specifies the interfaces file to use. If not specified, charset uses the interfaces file in the SYBASE directory.
-v	Causes the Sybase version string to be printed, then exits. Use with no other options specified.
sort_order_file | charset.loc	When charset is used to load the default character set and sort order, *sort_order_file* is a mandatory parameter specifying the name of the sort order file to be used by Adaptive Server. When loading additional character sets, use *charset.loc* to indicate the name of the character-set files.
charset_directory	Specifies the directory of the character set to be used by Adaptive Server.

CHAPTER 9

Upgrading Sybase Servers

This chapter describes how to upgrade approved versions of Adaptive Server, SQL Server, Backup Server, and Monitor Server to version 12.0.

Topics covered include:

Name	Page
Overview	139
Pre-Upgrade Tasks	140
Upgrading to Adaptive Server 12.0	153
Post-Upgrade Tasks	157
Upgrading Backup Server, Monitor Server, and XP Server	162
What's Next?	163

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

You can upgrade to Adaptive Server 12.0 from any of these versions:

- 11.0.x
- 11.5
- 11.5.1
- 11.9.2

The upgrade process includes:

- Performing generic pre-upgrade tasks to prepare your site environment and your current server for the upgrade
- Performing specific pre-upgrade tasks if your current server supports auditing or replicated databases

- Upgrading to the new release level
- Performing post-upgrade tasks

Note Before you begin the upgrade process, unload Adaptive Server 12.0 product files onto your system. The unloading process is described in Chapter 3, “Unloading Server Products from Distribution Media”.

Pre-Upgrade Tasks

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 your current Adaptive Server configuration, it may not be necessary to perform all pre-upgrade tasks.

Pre-upgrade tasks include:

- Check System and Upgrade Requirements.
- Note Server and Device Names and Locations.
- Test Current Applications and Stored Procedures.
- Check for Reserved Word Conflicts.
- Create a *sybsystemdb* Database.
- Verify That Users Are Logged Off.
- Disable Disk Mirroring.
- Check Database integrity.
- Back up Databases.
- Dump Transaction Logs.
- Turn Off Database Options.
- If you use replicated databases, read Appendix A, “Preparing to Upgrade Servers with Replicated Databases”.
- If applicable, disable auditing. See “Upgrading a Database That Uses Auditing” on page 152.

Check System and Upgrade Requirements

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 12.0.

To determine the release level of your current Adaptive Server (or SQL Server), enter the following command after login in to your server using **isql**:

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

You can upgrade to Adaptive Server 12.0 from any of these versions:

- 11.0.x
- 11.5
- 11.5.1
- 11.9.2

If you are running a version of Adaptive Server or SQL Server that is earlier than those listed here, you must upgrade to one of these versions before upgrading to 12.0.

3 Set the environment variables necessary to perform the upgrade to contain the necessary values as described in Chapter 5, “Installing Sybase Servers”.

4 Verify that you unloaded Adaptive Server 12.0 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 Reload 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 release level and has all operating system patches needed for Adaptive Server.

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 the old *$SYBASE/install* directory and is still named *RUN_server_name*, where *server_name* is the name of the current server. The *server_name* must appear exactly as it appears in the interfaces file.The runserver file for a server named SYBASE is named *RUN_SYBASERUNSERVERRUN_SYBASE*. If the runserver file for your current Adaptive Server is named, you must change the name to 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 (or SQL Server), the previously installed version of the server *must* be running.

Check for Reserved Word Conflicts

Check for reserved words that may conflict with identifiers currently in use by existing databases. 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 144. 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. Be sure to rename all objects that use reserved words.

See the *Adaptive Server Reference Manual* for a complete list of reserved words.

Note If you change an object name, be sure to 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.

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 *Adaptive Server 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 144 for more information.

When you perform a reserved word check with **sqlupgrade** or **sqlupgradres** it installs new reserved words and **sp_checkreswords**. While you address reserved word conflicts, you can run **sp_checkreswords** to verify that all conflicting names have been resolved. **sp_checkreswords** checks for all reserved words used as identifiers in the current database. Run this procedure 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 *Adaptive Server 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

Use the following procedure 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 boot the server in single-user mode.

Check Database integrity

Check logical and physical database consistency by running the following commands in each database:

- **dbcc checkalloc**
- **dbcc checkdb**
- **dbcc checkcatalog**

If you encounter any errors while running these commands, resolve them before continuing the upgrade. For help with resolving errors, see the *Troubleshooting and Error Messages Guide.*

If you are upgrading from Adaptive Server 11.5.1 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**.** See step 13.

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.

Database and Devices

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*.

Use **sphelpdb** to:

- Verify that the *tempdb* database is a minimum of 3MB.
- Verify that the *model* database is a minimum of 3MB.
- Verify that the master database is a minimum of 7MB.

If any database is smaller that the required size, use **alter database** to increase the size.

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. In version 12.0, all servers must have a *sybsystemdb* database. Version 12.0 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* from 11.5.x or 11.9.x and the database is at least 4 MB, 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* at the minimum database size (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% of the database should be data storage, and 75% 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


1> disk init name = "sybsystemdb_log"
2> physname = "/sybase/data/sybsystemdb_log"
3> vdevno = 9
4> size = 7680


1> create database sybsystemdb on sybsystemdb_data = 5
2> log on sybsystemdb_log = 15
```

sybsystemprocs

Verify that the *sybsystemprocs* database is large enough. The minimum size for *sybsystemprocs* is 80MB. You may need more space if you are adding user-defined stored procedures.

If your *sybsystemprocs* database is smaller than 80MB, and you have room on the device that contains this database, use the **alter database** command to increase the database size.

To determine whether the *sybsystemprocs* database is large enough, use **sp_helpdb** to determine the size of the *sybsystemprocs* database:

```
1> sp_helpdb sybsystemprocs
2> go
```

To determine whether the system procedures device is large enough, use sp_helpdevice to determine the size of the *sybsystemprocs* device:

```
1> sp_helpdevice sybsystemprocs
2> go
```

If the resulting information indicates that the *db_size* setting is less than 80MB, you must increase the size of *sybstemprocs*.

Increasing the Size of the *sybsystemprocs* Database

The recommended system procedures device (*sysprocsdev*) size is 80MB, which allows extra space that you may need for your stored procedures and accommodates any size increase in the *sybsystemprocs* database.

If your current *sybsystemprocs* database does not have at least 80MB available space, 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 with 80MB, 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.

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=35
2> go
```

In this example, “sysprocsdev” is the logical name of the existing system procedures device and “35” 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:

```
Couldn’t find enough space on disks to extend
database sybsystemprocs
```

If there is space available on another device, you can expand *sysprocsdev* to a second device, or initialize another device that is large enough. For instructions on creating a larger *sysprocsdev* device, see “Increasing Device and Database Capacity for System Procedures” on page 149.

3 Verify that Adaptive Server has allocated more space to *sybsystemprocs* by issuing the command:

```
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 *Adaptive Server 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.

To create a larger system procedures device (*sysprocsdev*):

1 Use the **drop database** command to drop the existing *sybsystemprocs* database, using the name you assigned to sysprocsdev:

```
1> sp_configure "allow updates", 1
2> go
1> use master
2> go
1> drop database sybsystemprocs
2> go
1> begin transaction
2> delete from sysdevices
3> where name = "sysprocsdev"
4> go
```

Note If this command affects only one row, commit the transaction. If it affects more than one row, roll back the transaction and call Sybase Technical Support.

---

2 Shut down Adaptive Server or SQL Server:

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

3 If your sysprocsdev device was a UNIX file, delete the file at the operating system level.

4 Start the server.

5 Log in to the server using **isql**.

6 Find another existing device that has 80MB 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/sysproc.dat",
4> vdevno = 9,
5> size = 40960
6> go
```

where *$SYBASE/* is the path to your system procedures device.

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 80MB (80 x 512 = 40960). For more information on **disk init**, see the *Adaptive Server Reference Manual*.

7 Create a *sybsystemprocs* database of the appropriate size on that device, for example:

```
1> create database sybsystemprocs on sysprocsdev =
80
2> go
```

8 Run the **installmaster** script in the *old* Sybase installation directory, as follows:

```
isql -Usa -Ppassword -Sserver_name-i/old_dir/scripts/installmaster
 -oinstallmaster.out
```

Turn Off Database Options

To turn off database options:

1 Run the **sp_helpdb** stored procedure to show the options that are set for each database.

2 Use the **sp_dboption** stored procedure 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, type:

```
sp_dboption sybsystemprocs, 'trunc log on chkpt', false
```

Be sure to turn off the **trunc log on chkpt** database option to avoid upgrade failure.

For more information about using these system procedures, see the *Adaptive Server Reference Manual*.

Note The database options for *master* cannot be changed and are not disabled.

Disable Disk Mirroring

Use the **disk unmirror** command to disable disk mirroring for your current Adaptive Server or SQL Server. For more information about this command, see the *Adaptive Server Reference Manual*.

Checking Memory Caches

Use **sp_configure** to check the amount of memory used by the following parameters:

- **procedure cache percent**
- **total data cache size**

Record the amount of memory and the procedure cache percent "Run Value" to use after you complete the upgrade. Memory use is given in kilobtyes by **sp_configure**.

Adaptive Server 12.0 uses more memory for code and internal structures than previous versions. If you have limited memory assigned for your Adaptive Server, you may encounter problems starting the new Adaptive Server during the upgrade process.

To avoid this problem, reconfigure your current Adaptive Server to provide sufficient memory for the upgrade process to complete successfully.

You can accomplish this task in either of two ways:

- Raise the memory allocated to Adaptive Server, or
- Lower other configuration parameters that consume memory.

  For example, use **sp_configure** to set user connections to the default value of 25. For more information about **sp_configure**, see the *Adaptive Server Reference Manual*.

  Make a note of the current values so that you can reset them after the upgrade.

Reboot Adaptive Server when you have finished configuring memory resources.

Upgrading a Database That Uses 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. Be sure to 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*.

Preparing to Upgrade with Replicated Databases

If you have a version of Replication Server running on a system that will use Adaptive Server version 12.0, follow the instructions in Appendix A, "Preparing to Upgrade Servers with Replicated Databases" *before beginning the upgrade process*.

Upgrading to Adaptive Server 12.0

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 153.
- **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 155.

Note Backup Server, Monitor Server, and XP Server cannot be upgraded.

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 release 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 release.

If the OLDSYBASE environment variable is set, **sqlupgrade** displays that value as the directory location for your current server.

7 Click Continue. The Adaptive Server Selection screen is displayed.

8 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.

9 Click Continue.

The SA Password screen is displayed.

10 Enter the password for the System Administrator (SA) login.

11 Click OK. The Specifying Upgrade Options screen is displayed.

12 Specify an upgrade option.

You can run the upgrade eligibility tests before performing the actual upgrade, or you can upgrade immediately after the upgrade eligibility tests run successfully.

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.

If **sqlupgrade** encounters database object names that use Transact-SQL keywords reserved by Sybase, it does not continue until you rename the database objects with names that are not reserved words.

13 Click OK to upgrade your server.

sqlupgrade displays the Status Output screen.

14 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*.

If the upgrade is successful, a “Done” message is displayed in the Status Output screen.

15 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 157.

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:

1 Use an ASCII text editor to edit the resource file as described in “Editing a Resource File” on page 156.

2 Execute **sqlupgraderes** using the edited resource file as described in "Upgrading using sqlupgraderes" on page 157.

Editing a Resource File

There are two ways to edit a resource file:

- Edit a sample resource file, or
- Edit the resource file created by the **sqlupgrade** utility.

For details on the attributes used in a resource file, see "Resource File Attributes for Upgrading Adaptive Server" on page 156.

Editing a Sample Resource File

A sample resource file is included in your Adaptive Server distribution in *$SYBASE/$SYBASE_ASE/init/sample_resource_files/upgrade.rs*. Use any ASCII editor to modify this file to specify the attributes for your upgrade.

Editing a Resource File Created by *sqlupgrade*

When you run the **sqlupgrade** utility, a new file containing the values you specified for that upgrade session is written to:

$SYBASE/$SYBASE_ASE/init/logs/sqlupgradeMMDD.NNN-server_name.rs

where *server_name* is the name of the upgraded server. You can create a new resource file by using an ASCII editor to modify this file.

Warning! Be sure to change device names in the resource file, if necessary, before executing the resource file using **sqlupgraderes**.

Resource File Attributes for Upgrading Adaptive Server

Table 9-1 on page 157 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*.

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 9-1: 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.

Verifying 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 “12.0” in the version string.

```
1> sp_configure "upgrade version"
2> go
```

sp_configure should return the Run Value “12000”.

Restoring 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 the upgrade to ensure enough memory, use the **sp_configure** command 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 12.0.

4 Determine the effect of upgrade on your caches.

When preparing for your upgrade, you recorded the amount of memory used by the **procedure cache percentage** and **total data cache size** configuration parameters.

Compare the previous values to the upgraded values to see what effect the upgrade had on your caches. Increase total memory, using **sp_configure,** to add enough memory to your Adaptive Server to raise your caches back to the previous values. This formula uses the old and new values for the procedure cache memory and the setting for **procedure cache percent** to determine how much memory to add:

```
select (old_proc_value + new_proc_value)/2 *
100/proc_cache_percent
```

Add the result to the current "Run Value" for **total memory**. You must reboot the server for this command to take effect. You may wish to add additional memory for the procedure cache, as explained in the next step, before you reboot.

For more information about **sp_configure**, see the *Adaptive Server Reference Manual*.

5 Check the procedure cache requirements.

The size of query plans has increased between releases. You may want to increase both the total memory and the percent of memory allocated to the procedure cache.

Stored procedures, triggers and other compiled objects require more memory to run than in older versions. The memory required to run a stored procedure increased by almost 100 percent between versions 4.9.x and 11.5 and approximately 20 percent between versions 10.x and 11.5. Adaptive Server 12.0 may need more procedure cache for the server to maintain the same performance.

The following table shows the space requirements for an upgrade.

Size increase of sysprocedures tables when upgrading to 12.0	Approximate % increase in procedure size:
From <10.x to 12.0 (32-bit)	74%
From <10.x to 12.0 (64-bit)	166%
From >=10.x to 12.0 (32-bit)	5%
From >=10.x to 12.0 (64-bit)	56%
From >=11.9.3x (64-bit) to 12.0 (64-bit)	5%

To increase the procedure cache size, increase the **total memory** or **procedure cache percentage** using **sp_configure**. Use **sp_sysmon** to monitor procedure cache usage.

For more information about **sp_configure** and **sp_sysmon**, see the *Adaptive Server Reference Manual* and the *Performance and Tuning Guide*. For more information about configuring memory, see the *System Administration Guide*.

6 If you unmirrored devices, remirror them, using the **disk remirror** command.

7 If you used two-phase commit in 11.5.x or 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.

Reenabling 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.

Removing 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
```

Reenabling 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 12.0, 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 9, “Upgrading Sybase Servers”.

Reenabling Auditing

Reenable all auditing options for stored procedures using the audit options you recorded during pre-installation. See “Upgrading a Database That Uses Auditing” on page 152). Reenter the audit options using **sp_audit**.

- Adaptive Server 12.0 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 *Adaptive Server 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

Backup Server, Monitor Server, and XP Server do not have a formal upgrade process.

Note Install a new Backup Server before you dump and load databases on the upgraded Adaptive Server. Back up all databases, including *master*, after successfully upgrading Adaptive Server and Backup Server.

To unload the new release of Backup Server, Historical Server, Monitor Server, and XP Server, follow the instructions in Chapter 3, “Unloading Server Products from Distribution Media”

To create the new Backup Server, and Monitor Server, follow the instructions in Chapter 5, “Installing Sybase Servers”

To create the new Historical Server, see the Historical Server documentation.

What’s Next?

You have completed the upgrade to Adaptive Server 12.0.

To localize your server, see Chapter 8, “Customizing Localization for Adaptive Server”.

To add extra functionality to your server, see Chapter 10, “Adding Optional Functionality to Adaptive Server”.

To configure and administer Adaptive Server, see the following sources:

- *System Administration Guide*
- *Performance and Tuning Guide*

CHAPTER 10

Adding Optional Functionality to Adaptive Server

This chapter provides instructions for adding the following optional functionality for Adaptive Server:

- Auditing – tracks security-related system activity in an audit trail, which can be used to detect penetration of the system and misuse of resources.
- Sample databases – French, German, Japanese, and U.S. English, databases that can be used to practice most examples given in the *Transact-SQL User's Guide*.

Topics include:

Name	Page
Installing Auditing	165
Installing Sample Databases	174

Installing Auditing

Auditing is an important part of security in a database management system. Security-related system activity is recorded in an audit trail, which can be used to detect penetration of the system and misuse of resources. By examining the audit trail, the System Security Officer can inspect patterns of access to objects in databases and can monitor the activity of specific users. Audit records can be traced to specific users, enabling the audit system to act as a deterrent to users who are attempting to misuse the system.

A System Security Officer manages the audit system and is the only user who can start and stop auditing, set up auditing options, and process audit data.

Audit System Devices and Databases

The audit system includes several components. The major components are:

- The sybsecurity device and the *sybsecurity* database, which stores audit information
- The audit trail, which is composed of several audit devices and tables that you determine at configuration time
- The syslogs transaction log device, which stores transaction logs

The *sybsecurity* Device and Database

The sybsecurity device stores the *sybsecurity* database. The *sybsecurity* database is created as part of the auditing configuration process. It contains all the system tables in the *model* database as well as a system table for keeping track of server-wide auditing options and system tables for the audit trail.

Tables and Devices for the Audit Trail

Adaptive Server stores the audit trail in system tables, named *sysaudits_01* through *sysaudits_08*. For example, if you have two audit tables, they are named *sysaudits_01* and *sysaudits_02*. At any given time, only *one* of the audit tables is *current*. Adaptive Server writes all audit data to the current audit table. A System Security Officer can use **sp_configure** to set or change which audit table is current.

When you configure Adaptive Server for auditing, you determine the number of audit tables for your installation. You can specify up to eight system tables (*sysaudits_01* through *sysaudits_08*). Plan to use at least two or three system tables for the audit trail and to put each system table on its own device, separate from the master device. If you do this, you can use a threshold procedure that archives the current audit table automatically, before it fills up and switches to a new, empty table for subsequent audit records.

Device for *syslogs* Systems Table

When you configure for auditing, you must specify a separate device for the *syslogs* system table, which contains the transaction log. The *syslogs* table, which exists in every database, contains a log of transactions that are executed in the database.

Overview of Audit Installation

There are two methods for installing auditing for the first time in Adaptive Server:

- Use the **installsecurity** script. For information, see the *System Administration Guide*.
- Use the **auditinit** utility. Tasks that you must perform before installing auditing and instructions on using the **auditinit** utility follow.

Pre-Installation Tasks

Determine the location of the raw devices you need for the *sybsecurity*, *syslogs*, and *sysaudits* table devices. You will need to provide this information later. For information on device selection, see "Pre-Installation Tasks" on page 54.

It is recommended that you configure your system with the minimum number of auditing devices you require—you must configure for at least three devices. You can add more auditing devices later with **sp_addaudittable**. For information, see the Adaptive Server *Reference Manual*.

Sybase recommends:

- Installing auditing tables and devices in a one-to-one ratio

  Tables that share the same device will share the same upper threshold limit. These tables cannot be used sequentially when a device fills up, because they both reside on the same device.

- Installing each auditing table on its own device

  This enables you to set up a smoothly running auditing system with no loss of auditing records.

  With two auditing tables, when one fills up, you can switch to the other. With a third auditing table, if one device fails, the System Security Officer can install a new threshold procedure that changes the device rotation to skip the broken device until the device is repaired.

- Making the device larger than the table

When you use only three auditing tables and devices, the size of the table and the size of the device can be similar, because you can obtain more auditing capacity by adding more auditing tables and devices (up to eight). When you are working toward the upper table and device limit (six to eight), you may want to make the device considerably larger than the table. Then, you can expand the table size later towards the upper size of the device when a larger auditing capacity is desired, and few or no device additions are available.

Audit Installation

To configure Adaptive Server for auditing:

1 Log in to your machine using your Sybase System Administrator ("sa") user account.

2 Start **auditinit** at the UNIX prompt:

```
$SYBASE/$SYBASE_ASE/install/auditinit
```

auditinit displays the following menu:

```
AUDITINIT
1. Release directory:  /usr/u/sybase
2. Configure a Server product
```

3 Select Configure a Server Product.

4 Select Adaptive Server.

5 Select Configure an Existing Sybase Server.

6 Select the server you want to configure.

7 Provide the SA Password for the server you selected.

8 From the Sybase Server Configuration screen, select Configure Auditing.

As you proceed through the series of menus in **auditinit**, you can change the default values, if provided. As you finish with each menu, press Ctrl-a to accept the defaults or changed values and move to the next menu.

```
CONFIGURE AUDITING
1. Configure auditing:  no
2. Add a device for audit table(s)
3. Add a device for the audit database transaction log
4. Delete a device entry
```

```
5. Change a device entry

List of devices for the audit tables:
Logical name    Physical name    Segment name    Table name    Size

Device for the audit datbase transaction log:
Logical name    Physical name    Segment name    Table name    Size
```

9 From the Configure Auditing screen, select Configure Auditing.

auditinit redisplays the Configure Auditing menu with the value “yes” displayed for Configure Auditing.

Creating a Device for an Audit Table

To create a device for an audit table:

1 From the Configure Auditing screen, select Add a Device for Audit Table(s).

auditinit displays the following menu:

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1. sybsecurity physical device name:
2. Logical name of the device:
3. Size of the device (Meg):
4. Device size for auditing:
```

2 Select Sybsecurity Physical Device Name.

3 Enter the *full path* of the physical device (raw partition) that you located in “Pre-Installation Tasks” on page 167.

```
Enter the physical name of the device to use for the audit database
(default is “”):
```

/dev/*path_to_partition*

where *path_to_partition* is the path to the raw partition for the device.

If you specify an operating system file, the following warning appears:

```
WARNING: '/secret1/sybase_dr/install/aud1.dat' is a regular file which
is not recommended for a Server device.
```

4 Press Return to acknowledge the warning.

auditinit redisplays the Add/Change a New Device for Auditing menu, which displays the physical name of the device:

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1. sybsecurity physical device
name:  /secret1/sybase_dr/install/aud1.dat
2. Logical name of the device:
3. Size of the device:
4. Device size for auditing:
```

5 Proceed through the remaining items on this menu. As you do so, be aware of the following:

The Size of the Device value must be equal to or greater than the Device Size for Auditing value. The Device Size for Auditing must be equal to the device size. If you are following Sybase auditing guidelines, you need not change the value displayed in Device Size for Auditing.

6 Press Ctrl-a to accept the settings. **auditinit** returns to the Configure Auditing menu and displays the device you have created.

```
CONFIGURE AUDITING
 1.  Configure auditing:  yes
 2.  Add a device for audit table(s)
 3.  Add a device for the audit database transaction log
 4.  Delete a device entry
 5.  Change a device entry

List of devices for the audit tables:
Logical name   Physical name    Segment name         Table name    Size

6.Audit_01' secret1/sybase_dr/install/aud1.dat' sysaudits_01 5
```

7 To add multiple audit devices, repeat steps 1–6.

You can add as many as eight devices. Sybase recommends adding three or more audit table devices.

After adding a device, **auditinit** returns to the Configure Auditing menu and displays all the devices you have created.

```
CONFIGURE AUDITING
1. Configure auditing:  yes
2. Add a device for audit table(s)
3. Add a device for the audit database transaction log
4. Delete a device entry
5. Change a device entry
```

```
List of devices for the audit tables:
Logical name    Physical name    Segment name         Table
name     Size

6. Audit_01'   /secret1/sybase_dr/install/aud1.dat’ sysaudits_01   5
7. Audit_02'   /secret1/sybase_dr/install/aud2.dat' sysaudits_02   5
```

Creating a Device for the Audit Database Transaction Log

To create a device for the audit database transaction log:

1 From the Configure Auditing menu, select Add a Device for the Audit Database Transaction Log.

auditinit displays the Add/Change a New Device for Auditing menu.

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1.  sybsecurity physical device name:
2.  Logical name of the device:
3.  Size of the new device (Meg):
4.  Device size for auditing:
```

2 Select Sybsecurity Physical Device Name.

auditinit prompts for the physical name and supplies you with a default, if available:

```
Enter the physical name of the device to use for the sybsecurity database
(default is''):
 /dev/path_to_partition
```

where *path_to_partition* is the path to the raw partition for the device.

3 Enter the full path name of a physical device.

If you enter an operating system file name, the following warning appears:

```
WARNING: '/secret1/sybase_dr/install/audlog'
 is a regular file, which is not recommended for a Server device.
```

4 Press Return to acknowledge this warning.

auditinit displays the Add/Change a New Device for Auditing menu and the value you selected for the physical name of the device.

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1.sybsecurity physical device name:
/secret1/sybase_dr/install/auditlog.dat
2.Logical name of the device:
3.Size of the device:
```

```
4.Device size for auditing:
```

5 Proceed through the remaining items on this menu. As you do so, be aware of the following:

- Sybase recommends a minimum size of 2MB for the size of the transaction log.
- **auditinit** displays the size in both Size of the Device and in Device Size for Auditing in the Add/Change a New Device for Auditing menu.
- The Device Size for Auditing default value is equal to the size of the device, based on the assumption that you may want to devote the entire device to log for the auditing task. If you want to use only a subset of the device, you can edit the Size of the Device value.

6 Press Ctrl-a to accept the settings displayed in the Add/Change a New Device for Auditing menu.

auditinit returns to the Configure Auditing menu and displays all the devices you have created.

```
CONFIGURE AUDITING
1. Configure auditing:  yes
2. Add a device for audit table(s)
3. Add a device for the audit database transaction log
4. Delete a device entry
5. Change a device entry

List of devices for the audit tables:
Logical name    Physical name    Segment name         Table
name     Size

6. Audit_01'   /secret1/sybase_  dr/install/aud1.dat’ sysaudits_01   5
7. Audit_02'   /secret1/sybase_  dr/install/aud2.dat' sysaudits_02   5
8. auditlog    /secret1/.../auditlog.dat logsegment   syslogs        2
```

7 When you are ready to execute the audit configuration, press Ctrl-a. **auditinit** returns you to the Sybase Server Configuration screen.

8 Press Ctrl-a again. **auditinit** prompts with:

```
Execute the Sybase Server Configuration now?
```

9 Enter “y” (yes).

auditinit executes the tasks to install auditing. When the installation completes successfully, the following messages are displayed:

```
Running task: install auditing capabilities.
.....................Done
Auditing capability installed.
Task succeeded: install auditing capabilities.
Configuration completed successfully.
Press <return> to continue.
```

Enabling Auditing

After auditing is installed, no auditing occurs until a System Security Officer enables auditing with **sp_configure**. For more information, see the *System Administration Guide*.

Deleting a Device Entry

To delete a device entry:

1 Select Delete a Device Entry from the Configure Auditing menu.

2 Enter the number of the device you want deleted.

3 Press Return to delete the entry.

Changing a Device Entry

To change a device entry:

1 Select Change a Device Entry from the Configure Auditing menu.

2 Enter the number of the device you want changed.

auditinit displays the Add/Change a New Device for Auditing menu with information on the device you selected:

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1. sybsecurity physical device name:
      /secret1/sybase_dr/install/audlog
2. Logical name of the device: aud.log
3. size of the new device (Meg): 5
4. Device size for auditing:5
```

3 Select each remaining entry you want to change.

4 Press Ctrl-a to save the new entries.

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:

Installing

- "Running the Database Scripts" on page 175
- "Installing the interpubs Database" on page 176
- "Installing the jpubs Database" on page 177

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

Table 10-1 lists the scripts that you can use to install the sample databases:

Table 10-1: 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. Be sure to run the installpix2 script after you run installpubs2.

Default Devices for Sample Databases

The *$SYBASE/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.

Each sample database requires 3MB on your database device. By default, the sample database installation scripts install the sample databases on the device that is designated as the default database device, which is the master device.

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. This configuration is not recommended because it uses valuable space that is best used for system tables.

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

- Use **sp_diskdefault** to specify a default device other than the master device. For information on **sp_diskdefault**, see the Adaptive Server *Reference Manual*.
- Modify each sample database installation script to specify a different device.

Running the Database Scripts

To run any of the scripts listed in Table 0-1:

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 **installpubs2** and **installpubs3** 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**.

5 Go to the *scripts* directory in your Sybase installation directory (*$SYBASE/$SYBASE_ASE*), and use the following syntax to 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
```

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

```
isql -Usa -Ppassword -Sservername\
<$SYBASE/ASE-12_0/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 the images 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.

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

To install *interpubs*:

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 -Jiso_1 \
 <$SYBASE/$SYBASE_ASE/scripts/iso_1/installintpubs
```

For more information on the **-J** option in **isql**, see the *Utility Programs for UNIX Platforms*.

jpubs Database

If you installed the Japanese Language Module with your Adaptive Server, *$SYBASE/scripts* 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 SunJLE 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 the EUC-JIS or Shift-JIS character set (depending on which characters you want for *jpubs*) 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 174.

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 \
 < $SYBASE/$SYBASE_ASE/scripts/eucjis/installjpubs
```

or:

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

For more information on the **-J** option in **isql**, see *Utility Programs for UNIX Platforms.*

Note Since you may want to refresh or make new copies of the sample databases, back up the original and edited versions of the installation scripts and store the copies in a safe place.

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.

What's Next?

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

CHAPTER 11

Starting and Stopping Servers

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

Topics covered include:

Name	Page
Overview	179
Starting Servers	180
Starting Servers When the Operating System Restarts	181
Stopping Servers	183

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*.

XP Server is not started by the installation process. XP Server is started only when any XP command is issued through **isql**.

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.

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

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.

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*.

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.

The runserver file for a server named SYBASE is named *RUN_SYBASE*. If the runserver file for your current Adaptive Server is named *RUNSERVER*, you must change the name to *RUN_SYBASE* during the upgrade process.

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 *Utility Programs for UNIX Platforms* 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 *Adaptive Server Enterprise Monitor Server User's Guide*.

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.

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 at operating system restart, see the *Sybase Adaptive Server Enterprise Monitor Server User's Guide*.

Before you set Adaptive Server or Backup Server to start automatically, make sure that all network resources are available. Your server will not start if the network is not up. Be sure that the entry for restarting the server *follows* any commands in the *rc* directory for starting network operations. Slow start-up of network operations can prevent servers from starting, even if commands are in the correct order in the *rc* directory. You may want to designate a period of time for your server to wait before starting by inserting a **sleep** command at the beginning of the linked runserver file you created.

To configure your operating system for automatic restart of Adaptive Server and Backup Server:

1 Create a start-up script file with contents like the following:

```
$SYBASE/$SYBASE_ASE/install/startserver
-f $SYBASE/$SYBASE_ASE/install/RUN_servername
```

where *$SYBASE/$SYBASE_ASE/install/startserver* is the full path name of the **startserver** utility, and *$SYBASE/$SYBASE_ASE/install/RUN_servername* is the full path name of the runserver file for the server.

2 Copy the script to the */etc/init.d* directory, using syntax like the following:

```
cp script_name /etc/init.d/script_name
```

3 Review the contents of the */etc/inittab* file to determine the default run level for your operating system. The **initdefault** entry specifies the default run level, which is usually 2 or 3.

4 Use the **ln** command to create a hard link from the runserver file to the appropriate *run control (rc)* directory, *rc#*, where # is the default run level obtained in step 2.

Use syntax like the following:

```
ln /etc/init.d/script_name /etc/rc#.d/S##script_name
```

When you type the command to create this link, add an uppercase "S" and a 2-digit sequence number preceding the name of the script file. "S" indicates a start-up file. The sequence number is necessary because files in the *rc* directory are executed in order of their numbers. This file should be executed last, so you should use a sequence number that follows all existing numbers in ASCII order.

You can perform an **ls** command on the *rc* directory to view existing sequence numbers. For example:

```
ls /etc/rc3.d/S*
```

returns the following results:

```
/etc/rc3.d/S10syslog


/etc/rc3.d/S15nfs.server


/etc/rc3.d/S21rfs
```

If your script is named **sybstart**, type the following:

```
ln /etc/init.d/sybstart /etc/rc3.d/S77sybstart
```

In this example, you can use any number greater than 21 in place of 77.

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 *sterr* file:

```
Server SHUTDOWN by request.The SQL Server is terminating
this process.
CT-LIBRARY error:
```

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 *Adaptive Server Reference Manual*.

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.

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 *Adaptive Server Enterprise 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 file in *$SYBASE* to store information about the shared memory segments that it uses:

SERVER_NAME.krg

When Adaptive Server is shut down in a normal manner, the shared memory files are automatically removed. If Adaptive Server crashes or is stopped with the **kill -9** command, these files are not deleted. You need read and write permissions on these files to restart Adaptive Server after a crash or a **kill -9** command, as Adaptive Server must be able to overwrite the previously created shared memory files.

CHAPTER 12

Troubleshooting

This chapter provides instructions for troubleshooting installation error messages.

If this chapter does not describe the error message you are encountering, see the *Troubleshooting and Error Messages Guide*.

Topics covered include:

Name	Page
Overview	187
Error Log Locations	188
Solutions to Common Installation Problems	189

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 188 for the location of the error log files for the installation utilities and the servers.

Table 12-1 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 are unable to correct the problem, see the *Troubleshooting and Error Messages Guide*.

Table 12-1: Troubleshooting guidelines

Problem	Possible Cause and Solution
The installation program cannot start Adaptive Server.	Failure to boot 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, unload all transfer 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.	See “Stopping Adaptive Server After a Failure” on page HIDDEN.
The upgrade fails.	See “Recovering from a Failed Upgrade” on page 194.

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 12-2 lists the default error log locations for the installation, upgrade, and setup utilities.

Table 12-2: 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 12-3 lists the default error log locations for each Sybase server.

Table 12-3: 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

Some common installation problems and possible solutions follow.

If this section does not describe the problem you are experiencing, see the *Troubleshooting and Error Messages Guide*.

You Do Not Have a 17-Inch Monitor

If you do not have a 17-inch monitor, you may need to change to a smaller font size to enable the X-Windows/Motif GUI installation and setup utilities (**srvbuild**) to display properly.

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 “Setting Environment Variables” on page 59.

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 Boot

The shared memory of the operating system may not be set high enough.

To correct the problem, see Chapter 5, “Installing Sybase Servers” for instructions on adjusting the shared memory value and start the installation or upgrade process again.

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.

Configuration Failed

Check the location of the system shared libraries.

The shared libraries *libXt.so* and *libX11.so* are normally stored in */usr/openwin/lib*. The shared library *libsocket.so* is normally located in */usr/lib*.

If the shared libraries on your operating system are located in directories other than those mentioned in the preceding paragraph, set the environment variable LD_LIBRARY_PATH to indicate the new locations.

See Chapter 3, “Unloading Server Products from Distribution Media” for instructions on how to set LD_LIBRARY_PATH.

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 12-3.

If Installation Fails After the Installer Creates Files

If the installation program quits while Adaptive Server is being configured, perform the following steps:

1 View the contents of the log file generated by Adaptive Server. For default error log locations, see Table 12-3.

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 boots the Adaptive Server that you are attempting to install, shut down that server. Follow the procedure under “Stopping Adaptive Server After a Failure” on page HIDDEN before performing step 5.

5 Use **asecfg** to restart the configuration.

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 ASE-12_0/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.

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.
```

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 148 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 12-3.

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.

- *First*, try to fix the problem that caused the upgrade to fail.
- *Then*, run the upgrade again.

If the upgrade fails:

- *Before* returning the message "Setting upgrade version to 12.0", you may need to restore your latest database backup, and restart the upgrade.
- *After* returning the message "Setting upgrade version to 12.0", 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, an 1105 error can usually be corrected 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 rerun 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 After a Failure” on page HIDDEN.

Shutting down the server enables the installation program to boot 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*.

CHAPTER 13

Configuring the Operating System

This chapter discusses the operating system configuration settings that you can adjust after installing or upgrading Adaptive Server. Unless stated otherwise, the information pertains to all supported UNIX platforms.

Topics covered are:

Name	Page
Setting Environment Variables	197
Using the stty Setting	199
Restoring Correct Permissions	199
File Descriptors and User Connections	199
Adjusting the Client Connection Timeout Period	203
Checking for Hardware Errors	203
Monitoring the Use of Operating System Resources	204
A Sample C Shell Maintenance Script	204
	206

Setting Environment Variables

It is crucial to the operation of Sybase products that the system environment variables are set correctly. 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).

Only the file owner or the "root" user can edit a user's *.login*, *.cshrc*, or *.profile* file.

You may want to verify that the Adaptive Server environment variables in Table 13-1 are set correctly.

To check the current value of environment variables:

- Enter the following command at the operating system prompt:

```
setenv
```

Table 13-1 describes the system environment variables for Adaptive Server.

Table 13-1: System environment variables

Variable	Sample Value	Function
DSLISTEN	IGNATZ	Used at Adaptive Server start-up; enables the specified Adaptive Server to listen for input from front-end software.
DSQUERY	IGNATZ	Defines the Adaptive Server to which local client applications connect
PATH	*/usr/bin:/etc: /usr/sbin: /usr/ucb: /usr/bin/X11:/sbin*	Appends the full path to the Adaptive Server bin subdirectories.

DSLISTEN

The DSLISTEN environment variable defines the name used by Adaptive Server to listen for client connections, when no name is given during Adaptive Server start-up. If DSLISTEN is not set, and no name is given during start-up, the Adaptive Server name defaults to the server name given at installation.

DSQUERY

The DSQUERY environment variable defines the Adaptive Server name to which client programs attempt to connect when no Adaptive Server name is specified with a command line option. If DSQUERY is not set, and you do not supply the Adaptive Server name with a command line option, clients try to connect to the server name given at installation.

PATH

The PATH environment variable specifies which directory paths to search for executables. The Sybase executables are in the *bin* and *install* directories in the Sybase installation directory. The install program appends these paths to the current PATH environment variable.

Note The new executables are added at the *end* of the PATH variable. Be sure to verify that you do not have any old Sybase executables, instead of the new executables, in the PATH that will be invoked.

Using the *stty* Setting

Setting the **stty tostop** option causes a background Adaptive Server to stop as soon as it tries to write to the terminal. To avoid this error, execute the following command before starting Adaptive Server:

```
stty -tostop
```

If you are redirecting all Adaptive Server output to files, you do not have to change the **stty** setting.

Restoring Correct Permissions

Sybase software files and directories are installed with the correct access permissions. If you notice that the permissions are no longer correct, you can restore the correct permissions with the script **setperm_all**, located in the *$SYBASE/$SYBASE_ASE/install* directory.

File Descriptors and User Connections

The number of user connections used by Adaptive Server cannot exceed the number of file descriptors available to Adaptive Server on the operating system. When configuring user connections on Adaptive Server, the System Administrator should take into account the number of file descriptors available per process. Although most of the open file descriptors are available for user connections, a few are used by Adaptive Server for opening files and devices.

Sample Program

The following example shows the source code that you can use to increase the hard limit:

```
#include <sys/time.h>
 #include <sys/resource.h>
 #include <sys/types.h>
 /*
 ** define MAX_CONNECTIONS to a number less than
 ** 10000. The number defined will then become the maximum
 ** number of connections allowed by an Adaptive Server.
 */
 #define MAX_CONNECTIONS 9999
 extern int errno;

 main(argc,argv)
 char **argv;
 {
     struct rlimit rlp;
     uid_t uid;

     rlp.rlim_cur = MAX_CONNECTIONS;
     rlp.rlim_max = MAX_CONNECTIONS;
 [nbsp  ]   /* set the number of open file desriptors to
     MAX_CONNECTIONS */
     if (setrlimit (RLIMIT_NOFILE,&rlp) == -1)
     {
        perror("setrlimit");
        exit(1);
     }

 [nbsp  ]   /* reset the user id to disable superuser
     privileges */
     uid = getuid();
     setuid(uid);
 [nbsp  ]   /* run the program indicated as arguments to
     this program */
     execv(*++argv, argv);
 }
```

For additional information on user connections, see the *System Administration Guide*.

For Sun Solaris

For Sun Solaris you can set both "soft" and "hard" limits for file descriptors. The soft limit can be increased up to the hard limit by the user, but the hard limit can be increased only by someone with "root" permissions. The soft limit determines the number of open file descriptors available to an Adaptive Server engine.

Although most of the open file descriptors are available for user connections, a few are used by Adaptive Server engines for opening files and devices.

See the *System Administration Guide* for additional information on user connections.

Displaying Current Soft and Hard Limits

To display the current soft limit:

- For C shells, enter:

  ```
  limit descriptors
  ```

- For Bourne shells, enter:

  ```
  ulimit -n
  ```

To display the current hard limit:

- For C shells, enter:

  ```
  limit -h descriptors
  ```

- For Bourne shells, enter:

  ```
  ulimit -Hn
  ```

Increasing the Soft Limit

To increase the soft limit:

- For C shells, enter:

  ```
  limit descriptors n
  ```

- For Bourne shells, enter:

  ```
  ulimit -Sn new_value
  ```

where *n* is the current value for the soft limit, and *new_value* is the value to which you want to increase the soft limit.

Note The preceding commands can be used in your *runserver* file to increase the hard and soft limits. Because the *runserver* file is a Bourne shell script, be sure to use the Bourne shell versions of these commands in the *runserver* file.

Increasing the Hard Limit

To increase the hard limit, use a program like the sample program shown in "Sample Program" on page 200.

To set up the sample program to increase the hard limit:

1. Create the ASCII file *file_name.c* (where *file_name* is the name you give the file), by using an ASCII text editor. Type the text shown in the sample in "Sample Program" on page 200.
2. Compile the file using the following command:

   ```
   cc file_name.c -o program_name
   ```

   where *file_name* is the name of the source file you created, and *program_name* is the name you want to give the program.
3. Change the program's permissions and ownership so that it will execute as "root":

   ```
   chmod 755 program_name
   chown root program_name
   ```

   where *program_name* is the name of the compiled program.
4. The "root" user can use the program to start Adaptive Server with increased user connections by typing the following command at the operating system prompt:

   ```
   # program_name dataserver -d master_device_name
   ```

   where *program_name* is the name of the compiled program, and *master_device_name* is the full path of Adaptive Server's master device. Instead of typing the command at the operating system prompt, you can add *program_name* preceding the dataserver command line in the Adaptive Server *runserver* file.

Adjusting the Client Connection Timeout Period

Adaptive Server uses the **KEEPALIVE** option of the TCP/IP protocol to detect clients that are no longer active. When a connection to a client is inactive for a period of time (the *timeout period*), the operating system sends **KEEPALIVE** packets at regular intervals. If it does not receive a response from the client machine for any of these packets, the operating system notifies Adaptive Server that the client is no longer responding. Adaptive Server then terminates the client's connection.

The **KEEPALIVE** default timeout period is 2 hours (7,200,000 ms). To display the current time value, use the command for your platform as shown in the following sections.

For Sun Solaris

To display the timeout value, enter:

```
/usr/sbin/ndd -get /dev/tcp tcp_keepalive_interval
```

To reduce the timeout period to 15 minutes (900,000 ms.), enter:

```
/usr/sbin/ndd -set /dev/tcp tcp_keepalive_interval 900000
```

Checking for Hardware Errors

The following types of hardware error messages indicate problems that may lead to database corruption:

- Disk read, write, or retry errors
- Timeouts
- System panics
- Memory problems of any type

For Sun Solaris

Check the */var/adm/messages* file on a regular basis. If any of the types of hardware errors described in the beginning of this section appear, use the Sun Microsystems diagnostic tool, **sundiag**, to check memory and disks. See the operating system documentation for more information.

Monitoring the Use of Operating System Resources

The *System Administration Guide* discusses maintaining the optimal number of Adaptive Server engines for your work-load and system configuration. To determine the optimal number, you need to monitor system and CPU usage.

For Sun Solaris

Sun Solaris supplies the following tools to help monitor performance:

- The **iostat** command reports the amount of I/O on terminals and hard disks and how CPU time is spent.
- The **vmstat** command monitors virtual memory usage.
- The **netstat** command monitors network status.
- The **ps** command gives you an accurate snapshot of accumulated CPU time and usage for individual processes. This can be very helpful in determining the dataserver-, engine-, and process-specific loading.
- The **time** command can be useful in determining the various user, system, and real-time resources used over a complete run.

For details about these tools, see your operating system documentation.

A Sample C Shell Maintenance Script

Running **dbcc** checks and performing database backups protect the integrity and recoverability of your Adaptive Server databases. The following sample C shell script calls several **isql** scripts to help you do this:

```
#!/bin/csh -f
```

```
if ( -e dbcc_mail.out) then
rm dbcc_mail.out
endif
foreach i (*.dbcc)
isql -Usa -Ppassword < $i > dbcc_out
if ( 'grep -c 'Msg 25[0-9][0-9]' dbcc_out' ) then
echo "There are errors in" $i >> dbcc_mail.out
cat dbcc_out >> dbcc_mail.out
else
echo "Backing up " $i:r >> dbcc_mail.out
isql -Usa -Ppassword < $i:r.backup
endif
end
mail -s "Backup Report" jjones < dbcc_mail.out
```

The first set of scripts (one for each database with a file name appended with *.dbcc*) runs **dbcc checkalloc** and **dbcc checkdb** for each database and sends the messages to an output file called *dbcc_out*.

For example, the script **master.dbcc** runs **dbcc** to check the *master* database:

```
dbcc checkalloc (master)
go
dbcc checkdb (master)
go
```

The C shell script then runs the **grep** command to find 2500-level error messages in the **dbcc** output. The results of the **grep** command go into an output file called *dbcc_mail.out*.

Next, the script invokes an **isql** backup script for each database for which no 2500-level errors occurred and adds the "Backing up *database_name*" line to *dbcc_mail.out*. For example, the script **master.backup** backs up the *master* database:

```
use master
go
dump database master to master_dump
go
```

You may want to add appropriate **dump transaction** commands to your scripts.

If there are 2500-level error messages, the script does not back up the database. At the end of the script, *dbcc_mail.out* is mailed to the System Administrator "jjones," who then has a record of fatal **dbcc** errors and successful backups.

You can tailor the sample shell and **isql** scripts to suit the needs of your installation.

To have the scripts execute automatically, edit the *crontab* file, and add an entry similar to this:

```
00 02 * * * /usr/u/sybase/dbcc_ck 2>&1
```

In this example, a C shell script called **dbcc_ck** executes every morning at 2:00 a.m.

APPENDIX A

Preparing to Upgrade Servers with Replicated Databases

If you are upgrading from a version of Adaptive Server prior to 11.5.1, transactions that were committed before an Adaptive Server or SQL Server database is upgraded to version 12.0 cannot be replicated after the upgrade because of a new transaction log format. Therefore, make sure that Replication Server® processes all committed transactions in the log before you upgrade the database.

Follow the instructions in this appendix *before beginning the upgrade process. If you are running version 11.5 or later, the information in this appendix does not apply.*

Note 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 12.0, 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 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

Use the **logmon** script located in the *$SYBASE/$SYBASE_ASE/upgrade* directory on the dataserver you are upgrading to drain the logs. You are prompted to start and stop Replication Servers and LTMs, as the logs are being drained.

Note Before running **logmon**, set your SYBASE environment variable to the location of your current SQL Server. You must execute **logmon** using the full path name of your current SQL Server installation. After running **logmon**, set the SYBASE environment variable back to the location for your new Adaptive Server.

The syntax for **logmon** is:

```
logmon -S server_name -U user_name -P password [-l file_name]
  [-help] [-version]
```

where *server_name* is the SQL Server to be upgraded and *file_name* is the file into which the LTM restoration scripts are to be written. You use this file to reenable replication after the upgrade, as described in "Restoring Functionality in Adaptive Server" on page 158.

The **-help** option displays the help screen, and the **-version** option displays the version of the **logmon** script.

When the transaction logs are drained, **logmon** displays the full path name of the file *logmon.server_name*, which you will need to reenable replication after upgrading.

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.

Disabling the Log Truncation Point

When you upgrade a primary database, the Log Transfer Manager (LTM) must not be running, and the LTM truncation point should be turned off for the duration of the upgrade.

Before upgrading the databases:

1 Shut down the LTMs for the databases that you are upgrading.

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

3 In each primary database and primary RSSD, execute the following commands to turn off the LTM truncation point:

```
1> use database
2> go
1> dbcc settrunc("ltm", "ignore")
2> go
```

If the **dbcc settrunc** command fails because the server "context" is reserved, execute the following commands to allow the server to recognize that the LTM is disconnected:

```
1> begin tran
2> commit tran
3> go 500
```

Then turn off the LTM truncation point:

```
1> dbcc settrunc("ltm", "ignore")
2> go
```

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

Draining Each Replicated Primary

Manually drain the transaction log of each replicated primary in the data server you are upgrading:

1 Note the timestamp of the last log page:

```
1> use database
2> go

1> dbcc traceon (3604)
2> go
```

This sends trace output to the terminal.

```
1> select root from sysindexes where name = 'syslogs'
2> go
```

The value returned is the last log page.

```
1> dbcc page (<database>, <last_log_pageid>, 0)
2> go
```

The timestamp is a 6-byte hexadecimal number. Save this value for reference in step 3.

2 Append to the log:

```
1> use database
```

```
2> go

1> begin tran
2> commit tran
3> go 500
```

3 Check the LTM truncation point:

```
1> use database
2> go

1> dbcc gettrunc
2> go
```

The value in the *ltm_trunc_page* column is the LTM truncation point.

```
1> dbcc traceon (3604)
2> go
```

This forces trace output to the terminal.

```
1> dbcc page (<database>, <ltm_trunc_page>, 0)
2> go
```

Note the timestamp. Continue this step until the timestamp of the LTM truncation page is greater than the timestamp value you saved in step 1.

Shut down the LTM for this replicated database, and then disable the LTM truncation point.

```
1> use database
2> go

1> dbcc settrunc ('ltm', 'ignore')
2> go
```

Suspend the connection to the RSSD databases from Replication Server.

```
1> suspend connection to dataserver.database
2> go
```

Restoring Replication After Upgrade

Restore the replication functionality:

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

2 Using **isql**, connect to the RSSD and execute the following commands:

```
1> use RSSD
2> go

1> rs_zeroltm dataserver, RSSD
2> go
```

3 Set the LTM truncation point to “valid” in each replicated primary. Using **isql**, connect to the replicated primary, and execute the following commands:

```
1> use <database>
2> go

1> dbcc settrunc (‘ltm’, ‘valid’)
2> go
```

Upgrading Adaptive Server

Follow the instructions in Chapter 9, “Upgrading Sybase Servers” to complete the upgrade of your replicated databases.

Index

A

Accented letters 121
Adaptive Server
  base license 38
  character set, changing 72
  character sets 117
  clients 3
  communicating with Backup Server 66
  conversions between, and clients 117
  database device requirements 56
  default character set 72
  default sort order 72
  described 2
  disk space requirements 17
  installation directory structure 7
  installing multiple 73
  installing stored procedures for metadata 84
  installing via resource file 73
  language, changing 72, 78
  localizing 62, 72
  minimum platform requirements 54
  more information about 77
  network protocol for 15
  performance statistics for 3
  preparing to upgrade 144
  RAM required by 15
  shutting down 183
  sort order 72
  starting from UNIX command line 180
  starting with operating system 181
  start-up script 182
  support for international systems. See Localization 109
  System Administrator 79
  total memory parameter 55
  verifying environment for upgrade 141
Adaptive Server Anywhere
  installing stored procedures for metadata 84
Adaptive Server Enterprise
  installing stored procedures for metadata 84
Adaptive Server Plug-in for Sybase Central 96, 97
Additional licenses
  SySAM Certificate 45
Administrator
  operating system 2
  Sybase system 2, 23
Advanced Security Mechanism
  authenication and encryption 38
  encryption using DCE and CyberSafe 38
Advanced Security Mechanisms
  enabling 48
alter database command 147
Applications
  changing after renaming objects 143
  testing current 142
Arabic character sets 113
Asynchronous I/O (AIO)
  enabling 55
Audit system 165
Audit trail
  overview 165
  system audit tables 166
Auditing
  database for 166
  device for 166
  feature 161
  global options 166
  installing using the auditinit utility 167
  installing using the installsecurity script 167
  process 166
  recommended database device size 56
  tables for tracking 166
auditinit utility 12
Authenication
  encyrption 38
Authentication
  advanced security mechanism 38
Authorization code
  troubleshooting 50

B

Backup Server
character set for 65
character sets 117, 132
communicating with Adaptive Server 66
configuring 127, 131
default language for 65
described 3
when to install 3
installing via resource file 73
more information about 78
network connections for 65
server connections 66
specifying message language for 65
starting from UNIX command line 180
starting with operating system 181
using tape with 3
Base license, Adaptive Server 38
Batch installation of servers 73
bcp utility 92, 111
Binary sort order 72, 121
Bourne shell 197
Browser
Netscape restrictions on CLASSPATH 83
Buffer specifications 21

C

C shell 197
CD-ROM
eject command 32, 34
Changing
character sets 72, 131
languages 131
server language 72
sort order 131
Character devices
located in /dev directory 57
verifying 57
Character sets 117
changing 72, 112, 131
client selection of 111
code conversions and 116
configuring 132
converting between 116, 117
databases and 120
default 111
for Backup Server 65
in a heterogeneous environment 116
sort orders and 120
charsets directory 120, 127
about the 128
Chinese character sets 113
CLASSPATH environment variable 81
Netscape restrictions on 83
Client products
installing 35
Clients
applications and locales.dat file 129
communicating with Adaptive Server 202
conversion between, and server 117
default character set 111
Code conversion
between character sets 117
Collating sequences. See Sort orders 120
Commands
iostat 204
netstat 204
ps 204
setenv 197
setperm_all 199
time 204
vmstat 204
common.loc file 128
compare function 124
Configuration file, network 106
Configuring
Backup Server 127, 131
character sets 132
Conflicting object names 143
Connection problems, analyzing 54
Connectivity 92, 104
Conventions, used in this manual xiii
Conversions, Unicode character 112
Converting between character sets 116, 117
create database command, system tables created by 11
.cshrc file 197
Current applications, testing 142
CyberSafe
advanced security mechanisms 38
encryption and authentication 38

Cyrillic character sets 114

D

Data translation 109
Database devices 174
  See also individual device names 56
  master 11
  raw partitions for, preparing 57
  sysprocsdev 11, 12
DatabaseMetaData methods
  stored procedures for 83
Databases 120
  dbccdb 13
  image data for sample 176
  interpubs 176
  jpubs 177
  master 11, 12
  model 11
  optional 174
  sample 13, 20, 178
  sizes of 20
  specifications 19
  suspect 145
  sybsecurity 12
  sybsystemprocs 11, 12
  tempdb 11
Databases interpubs 20
Databases jpubs 20
Databases master 20
Databases model 20
Databases pubs2 20
Databases pubs3 20
Databases sybsystemprocs 20
Databases tempdb 20
dataserver utility 185
Datatypes
  Hankaku Katakana and 119
  varbinary 122
dbcc checkalloc command 145
dbcc checkcatalog command 145
dbcc checkdb command 145
dbcc checkstorage, database for 13
dbcc checkverify command 145
dbcc error messages 205
dbccdb database 13
DCE
  advanced security mechanisms 38
  encryption and authentication 38
Dec-Kanji character set 117
Default
  character set for Adaptive Server 72
  character set for Backup Server 65
  character set, changing 72
  character sets installed by 72
  language for Adaptive Server 72
  language for Backup Server 65
  language, changing 72
  network protocols 104
  sort order 72
defncopy utility 111
Dictionary sort orders 121
Digital UNIX
  hardware error messages 203
  monitoring systems 204
Directories
  charsets 120, 128
  installation 7
  localization 127
  run control 182
  specifying new, for installation 141
disk init command 57
Displaying current file descriptors 199
Distributed Transaction Management
  enabling 48
  recovery 38
Documentation
  Adaptive Server translated 110
Documents
  Historical Server 35
  Monitor Server 35
Drivers, ODBC 93, 107
dscp utility 5, 10
dsedit utility 5, 10, 161
DSLISTEN environment variable 6, 80
  introduction to 198
DSQUERY environment variable 6, 80
  introduction to 198
dump database command, backing up with 145
Dump requests 3
dump tran command 145

E

Enabling
  monitoring functions 98
Enabling asynchronous I/O 55
Enabling features 48
  advanced security mechanisms 48
  distributed transaction management 48
  high availability 48
  Java 48
Encryption
  advanced security mechanisms 38
  authentication 38
Environment variables
  CLASSPATH 81
  DSLISTEN 6, 80, 198
  DSQUERY 6, 80, 198
  JDBC_HOME 81
  PATH 6, 80, 97, 199
  SYBASE 6, 7, 80
Environment variables, setting
  JAVA_HOME 25
Error log files, troubleshooting and 193
Errors
  resolving 145
Errors in dbcc messages 205
EUC-JIS character set 117
Extended stored procedures (ESPs) 4, 78

F

Failover
  high availability 38
Feature licenses
  adding additional licenses 45
  troubleshooting 50
Features
  licensed through SySAM 37
File descriptors
  displaying current 199
  hard limit 202
  soft limit 202
Files
  .cshrc 197
  .login 197
  .profile 197
  common.loc 128
  locales.dat 128
  localization 110
  localized error messages (.loc) 128
  runserver 180
  shared memory 186
  sort order definition (.srt) files 120
Formatting for local date, time, and currency 128
French sample database 13
Functions
  compare 124
  softkey 122

G

German sample database 13
Globalization support, Sybase 109, 127, 131
Greek character sets 114
Guest user
  in pubs2 and pubs3 databases 178

H

Hankaku Katakana conversions 118
  limitations on datatypes 119
Hard limit for file descriptors 202
Hardware error messages 203
  Digital UNIX 203
Hardware requirements 19
Hebrew character sets 114
Heterogeneous environments 112, 116
High Availability
  Adaptive Server failover 38
High availabiltiy
  enabling 48
Historical Server
  configuring and using 35
  description of 3
  documentation 35
Host name 66
  License Manager Host 31, 45
  primary network node 43

I

IBM RS/6000
  timeout period 203
Identifiers, conflicting with reserved words 143
InfoMaker
  described 94
Installation
  contents 84
Installation directory
  for unloading products 24
  structure of 7
Installation utilities 5
Installing
  jConnect 84
  multiple Adaptive Server versions 80
  stored procedures 83
  Sybase products 79, 175
Installing 11.9.2
  Adaptive Server Plug-in for Sybase Central 96, 97
  Open Client/C for Windows NT/95 101
Installing client products 35
Installing Java 98
Installing Sybase products 91
installintpubs script 176
installjpubs script 177
installpix script 176
installpubs2 script 175
installpubs3 script 175
Interfaces file 9
interfaces file 161
Interfaces File Entry attribute for custom installation 66, 69
International systems
  support for 109
  Sybase support for 109
interpubs sample database 13, 176
iostat command
  Sun Solaris 204
iso-Latin1 character set 72
isql
  checking for replicated databases 208
  determining Adaptive Server release level 141
  verifying which servers are running 80
isql utility 111

J

Japanese sample database 13
Java
  enabling 48
  in Adaptive Server databases 38
  XML classes and 38
Java Development Kit 98
Java Development Kit. See JDK, installing the 24
Java Runtime Environment 98
JAVA_HOME environment variable 25
jConnect
  installation contents 84
  packages 85
  testing the installation 84
JDBC 1.x samples 87
JDBC 2.0 samples 87
JDBC_HOME
  environment variable 81
  setting 81
JDK (Java Development Kit), installing the 24
JDK 1.1 85
jpubs sample database 13, 177

K

KEEPALIVE option, TCP/IP 203
keywords, reserved by Sybase 155
kill command 185, 186
Korean character sets 115

L

Language functions, procedural 4
Language Modules 93, 110, 126, 127
  description of 4
  disk space required for 17
  for Backup Server 65
  installing new 126
  localization files 110
  memory requirements for 131
  sizes of 18
Languages
  changing 131
  error reporting in specific 128

selecting message 127
translation support 109
Latin character sets 114
Letter case in sort orders 121
License files 38
sample license file 51
License management daemons
lmgrd 39
lmutil 39
License Manager Host, see host name 31, 45
License Manager Port 31, 45
License Manager Port, see port number 31, 45
Licensed features 37
Licenses management daemons
SYBASE 39
Limits for file descriptors 201
Listen port number, finding 106
lmgrd daemon
starting the 44
SYBASE daemon 44
lmreread
upgrading licenses 47
Load requests 3
loc files 128
Local date, time, and currency formatting 128
locales directory 127
locales.dat file 128
Localization 109
changing languages 72
changing the configuration 131
common, information 128
disk space needed for 62, 65
Login
administrator, Windows NT 94
xd2 rootxd3 2
xd2 saxd3 2
superuser 2
xd2 sybasexd3 2, 23
.login file 197
logmon command 209
Loopback mode 54

M

Mapping NLWNSCK to TCP 106
master database 11
Master device 11, 56
Messages
hardware errors 203
Messages, selecting language for 127
Metadata, stored procedures for 83
model database 11
Monitor Client Library
configuring and using 35
description of 3
RAM required by 17
Monitor Historical Server. See Historical Server 18
Monitor Server
described 3
description of 3
disk space required for 17
monserver command 181
more information about 78
shutting down 184
starting from UNIX command line 180
starting with operating system 181
Monitoring
operating system resources 204
Monitoring systems
Digital UNIX 204
monserver command 181
Multiple Adaptive Server products
SySAM support for 38
Multiple servers
installing in network asset management 42

N

Named Pipe network protocol 92, 104
Net-Library 92
Netscape
restrictions on CLASSPATH 83
netstat command
Sun Solaris 204
Network asset management
installing multiple servers 42
Network configuration file 106
Network connections
Backup Server 65
between clients and servers 104

Monitor Server 68
Network problems, and Adaptive Server data integrity 56
Network protocols
DECnet 10
Named Pipes 104, 106
SPX 10, 15
TCP 106
TCP/IP 10, 15
Winsock 104, 106
Network software configuration, verifying 54
Network-mounted devices 56

O

Objects
conflicting names of 143
renaming 143
ODBC 93, 107
Open Client applications 4
Open Client disk space requirements 17
Open Client/C
for Windows NT/95 92, 101
Open Database Connectivity. See ODBC 107
Open Server applications 3
Operating system
administrator 2
kernel patches 54
RAM requirements 15
resources 204
starting servers 181
Optional databases 174
Optional features
authorization code input error 50
enabling all 49
sample license file 51
turning on and off 48
order by clause 122
Ownership, of files and directories 23

P

Packages, jConnect 85
Parent process identification (PPID) 186
Partition map, avoiding damage to 57
Password, setting System Administrator 79
PATH environment variable 6, 80, 97, 199
Permissions 199
required in environment variables 197
restoring of 199
PID. See Process identification 186
Platform-specific locale names 128
plug-in for Adaptive Server 96
Plug-ins
installation requirements for 18
Port number 31, 43, 45
Port number, specifying 66
PowerDesigner, described 94
PowerDynamo
described 93
PPID. See Parent process identification 186
Preparing
Adaptive Server for upgrading 144
Primary network node
host name 43
port number 43
SySAM 40
primary network node 43
Privileges 58
Procedural language functions 4
Procedure specifications 21
Process identification (PID) 186
.profile file 197
ps command
Sun Solaris 204
pubs2 sample database 13
pubs3 sample database 13

Q

Query specifications 21
Quick start installation
quick test 64
Quick test
quick start installation 64

R

RAM, requirements for
client products 17
servers 15
Raw partitions 56
availability of 58
choosing for database devices 57
minimum size for database devices 57
Recovering from a failed installation 193
Recovery
distributed transaction management 38
Reenable Replication Server 160
Referential integrity constraint 21
Removing
all products 89
Renaming objects 143
Replicated databases
checking for 208
draining 211
restoring 212
Replication Server 160
draining logs for Adaptive Server 160, 161, 209
Reserved word conflicts
checking for 144
Reserved words
checking for conflicts 144
using quoted identifiers 144
Resource files 73, 155
installation using 73
roman8 character set 72
Run control directory 182
Runserver file 180

S

xd2 saxd3 user 79
Sample databases 174, 178
Samples
JDBC 1.x 87
JDBC 2.0 87
Scripts
C shell 204
installintpubs 176
installjpubs 177
installpix 176
installpubs2 175
installpubs3 175
maintenance 204
sample maintenance 204
start-up 182
Security. See Auditing 165
select command 122
Server connections for Backup Server 66
Servers, including in sql.ini files 104
setenv command 197
setperm_all command 199
Setting, JDBC_HOME 81
Setup utilities 5
Shared memory files 186
Shared memory parameters, setting 54
Shells
Bourne 197
C 197
Shift-JIS character set 117
showserver command 80, 158
shutdown command 183, 184
Size
interpubs database, minimum 20
interpubs database, minimum for upgrade 20
jpubs database, minimum 20
jpubs database, minimum for upgrade 20
language module 18
master database, minimum for new installation 20
master database, minimum for upgrade 20
model database, minimum for new installation 20
model database, minimum for upgrade 20
pubs2 database, minimum 20
pubs3 database, minimum 20
pubs3 database, minimum for upgrade 20
recommended for database devices 147
sybsystemprocs database, minimum required for upgrade 20
tempdb database, minimum for new installation 20
tempdb database, minimum for upgrade 20
slloc utility 120
Soft limit for file descriptors 202
softkey function 122
Sort order
Adaptive Server default 72
changing 72
default for Adaptive Server 72

Sort orders 120
binary 121
changing 112, 131
character collation behavior 122
character sets and 120
collation behavior 122
collation operations 122
databases and 120
definition files 120
dictionary 121
letter case in 121
sp_configure
enable all features at boot up. 49
SPX network protocol 10, 15
SQL Anywhere
installing stored procedures for metadata 84
SQL Modeler. See PowerDesigner 94
SQL Remote
described 94
SQL Server
Monitor Server for version 11.0.x 3
sql.ini file 104
sqledit utility 92
sqlloc utility 5
sqllocres utility 5
sqlupgrade utility 5, 155
sqlupgraderes utility 5, 155
srt files 120
srvbuild utility 5, 10, 60, 72
srvbuildres utility 5
Starting servers
from UNIX command line 180
with operating system 181
startserver utility 180
Stored procedures
Adaptive Server Enterprise 84
changing after renaming objects 143
recommended database device size if adding user-defined 147
required for metadata 83
scripts for installing 83
SQL Anywhere or Adaptive Server Anywhere 84
testing 142
stty settings 199
Sun Solaris
iostat command 204
netstat command 204
ps command 204
time command 204
timeout period 203
vmstat command 204
sundiag system diagnostic tool 204
Suspect databases 145
Sybase Central 96
SYBASE environment variable 6, 7, 80
Sybase globalization support 109, 127, 131
Sybase installation directory 7, 24, 25
Sybase products
described 93
InfoMaker 94
installing 79, 91, 175
Language Modules 93
Net-Library 92
Open Client/C for Windows NT/95 92, 101
PowerDesigner 94
PowerDynamo 93
running during installation 142
uninstalling 89
sybsecurity
database 12, 166
device 12, 56
sybsetup utility 5
sybsystemdb
device for 56
sybsystemprocs
database 12
sybsystemprocs database 11
preparing for upgrade 149
Syntax
conventions for, in this manual xiv
SySAM 38
adding component licenses 42
base product 41
base product and licensed features 41
configuration 40
defined 37
features licensed through 37
functionality tracking 41
licenses files 38
manually enabling 42
multiple license files 38
Network asset management 42

networked servers 40
primary network node 40
standalone 40
starting manaully 44
support for multiple Adaptive Server products 38
troubleshooting 49
types of asset management systems 41
SySAM Certidicate
sample license file 51
SySAM Certificates
adding additional licenses 45
SySAM standalone 41
sysprocsdev device
minimum size for 56
purpose of 11, 12
sysprocsdev device, preparing for upgrade 149
System Administrator
setting password 79
System audit tables 166
System databases
where stored 56
System messages, translated 110
System procedures
storage location of 12

T

Table specifications 21
Tape, using with Backup Server 3
TCP/IP 203
KEEPALIVE option 203
TCP/IP network protocol 10, 15, 106
tempdb database 11
Testing for a successful jConnect installation 84
Thai character sets 115
time command
Sun Solaris 204
Timeout period for client connections 202
Transaction log, space in 145
Translated messages
error (.loc files) 128
system 110
Transport type menu 66
Troubleshooting
by using error log files 193
cannot run monitoring in Sybase Central 98
common installation problems and solutions 188
feature licenses 50
SySAM 49
SySAM authorization codes 50
Turkish character sets 115

U

Unicode
character conversion 112
Uninstall option 89
Uninstalling Sybase products 89
UNIX
system requirements 16
Unloading
11.5.1 Monitor Server for SQL Server 11.0.x 25
server products 23
Upgrading licenses
lmreread 47
us_english language 72
User connections 199
Utilities
bcp 92, 111
defncopy 111
dsedit 92, 161
isql 111
slloc 120

V

varbinary datatype 122
Version program, testing for successful jConnect installation 84
vmstat command
Sun Solaris 204

W

Windows 95 and NT
client product installation on 94
plug-in installation on 18
Windows Socket network protocol 92, 104

X

XML classes
  and Java classes in 38
XP Server
  described 4
  more information about 78
  starting 179
  starts automatically with first ESP request 100
  unloading 27, 33, 94
xp_cmdshell command 179

Z

Zenkaku Katakana 118