

SYBASE®

Installation Guide

Adaptive Server® Enterprise

15.5

Linux

DOCUMENT ID: DC30119-01-1550-02

LAST REVISED: June 2010

Copyright © 2010 by Sybase, Inc. All rights reserved.

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

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

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

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

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

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

IBM and Tivoli are registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

All other company and product names mentioned may be trademarks of the respective companies with which they are associated.

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

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

Contents

Adaptive Server® Enterprise Architecture	1
Adaptive Server® Enterprise Editions	1
Determining the Current Edition	1
Options for Adaptive Server Editions	2
Installation Workflow	3
Style Conventions	3
Server Component Descriptions and Directory Layout	4
PC-Client Product Descriptions and Directory Layout	7
Planning the Installation	9
Obtaining a License	9
Accessing SPDC	10
Installing a New License Server	13
System Requirements for Servers	17
Sybase Support for OLE DB and ODBC	20
Memory Requirements	20
System Requirements for Clients	23
Preinstallation Tasks for the Server Installation	25
Obtaining the SySAM Host ID	28
Preinstallation for an Overlay Installation	28
Installing Server Components	31
Installing in GUI mode	31
Installing From the Command Line	37
Response File Change	37
Creating a Response File	37
Installing Server in Silent Mode	38
Installing Adaptive Server Using Binary Overlay	38
Determining the Adaptive Server Version	40
Backing up the Server	40

Upgrading an Adaptive Server Configured with High Availability	40
Monitoring and Diagnostic (MDA) Tables	41
New Monitoring and Diagnostic Tables	41
Monitoring Remote Servers Using MDA Tables ...	41
Removing Adaptive Server	42
Uninstalling	42
Removing an Existing Adaptive Server	43
Installer Changes from Adaptive Server 12.5.4 to Adaptive Server 15.5	44
PC-Client Installation	47
System Requirements for PC-Client	48
Installing the Client	48
Creating a Response File	49
Installing the Client in Silent Mode	50
Getting Started After Installing	50
Upgrade	53
Preparing to Upgrade	53
Preupgrade Tasks	55
Reserved Words	56
Preparing the Database and Devices for the Upgrade	58
Increasing Default Database Sizes	59
Upgrading to Adaptive Server 15.5	60
Using Sqlupgrade	61
Using Sqlupgraderes	62
Upgrading Using Sqlupgraderes	63
Postupgrade Tasks	64
Running the Instmsgs.ebf Script	65
Restoring Functionality in Adaptive Server	65
Reenabling Replication Server After the Dataserver Upgrade	67
Restore Replication on Destination Databases ...	67
Restoring Replication on Primary Databases	68
Reenabling Auditing	69

Special Considerations for Threshold	
Procedures for Audit Segments	69
Upgrading Backup and Monitor Servers	70
Upgrading Java in the Database	71
Enabling the Java in The Database Feature in a	
High Availability System	71
Upgrading Job Scheduler	71
Upgrading Job Scheduler Templates	72
Upgrading High Availability and Cluster Support	76
Migrating from 32-bit to 64-bit Versions on Different	
Computers	77
Migrating Data Using a Dump and Load Method	
.....	78
Using BCP to Migrate Data	78
Migrating Data by Replacing the Binary	78
Upgrading Using Allrows_dss In Parallel	79
Recovering From a Failed Upgrade	79
Downgrading from Adaptive Server 15.5	81
Post-downgrade Tasks	83
Additional Steps Required if You Used New Features ..	84
Downgrading Adaptive Server with Replicated	
Databases	85
Downgrading Job Scheduler	85
If You Downgraded Back to the Original Installation	
After Upgrading to 15.5	86
Troubleshoot SySAM	87
Where to Look For SySAM Related Errors	87
Troubleshooting SySAM	88
Enabling and Changing E-mail Notifications	94
Calling Sybase Technical Support	95
Troubleshoot the Server	97
Logs for Installation Utilities	98
Logs For Sybase Servers	99
Solutions to Common Installation Problems	99
Stopping Adaptive Server After a Failure	102

Recovering from a Failed Installation	102
If Adaptive Server Fails the Preupgrade	
Eligibility Test	103
Recovering from a Failed Upgrade	103
Upgrading Compiled Objects with Dbcc	
Upgrade_Object	105
Finding Compiled Object Errors Before	
Production	106
Syntax for Using Dbcc Upgrade_object	109
Syntax Examples for Upgrading Compiled	
Objects	110
Increasing the Log Segment Size	111
Upgrading Using Dump and Load	111
Upgrading Compiled Objects in Database	
Dumps	111
Raw Partitions on Linux	113
Choosing a Raw Partition	113
Creating Raw Partitions	114
Red Hat Raw Device Administration	115
SuSE Raw Device Administration	116
Accessing Raw Devices from the Server	116
Upgrading Servers with Replicated Databases	119
Determining If Existing Servers Contain Replicated	
Databases	119
Suspending Transaction Processing and Replication	
Activities	120
Draining the Transaction Logs for Primary Databases	
.....	120
Draining the RSSD Transaction Log	122
Disabling the Secondary Truncation Point	123
Using Sybssystemprocs	125
Increasing the Size of the Sybssystemprocs Database	
.....	125
Increasing Device and Database Capacity for System	
Procedures	126

Obtaining Help and Additional Information	129
Index	131

Adaptive Server® Enterprise Architecture

Adaptive Server Enterprise is based on the client/server model. Adaptive Server communicates with its clients over the network via the Tabular Data Stream™ (TDS) protocol. The client process may execute on one machine and communicate with a database server on the same or a different machine.

Adaptive Server runs as an application on top of an operating system. The hardware that the operating system runs on is completely transparent to Adaptive Server which sees only the operating system's user interfaces. You can configure multiple processes (engines) to enhance performance on multiprocessor systems.

Adaptive Server is divided into two components, a DBMS component and a kernel component. The kernel component uses the operating system services for process creation and manipulation, device and file processing, and inter-process communication. The DBMS component of Adaptive Server manages the processing of SQL statements, accesses data in a database, and manages different types of server resources.

Adaptive Server® Enterprise Editions

Adaptive Server editions and optional features are unlocked by SySAM licenses.

- Enterprise Edition – Has no limit on scalability and can run all available options that can be purchased separately
- Small Business Edition – Has limited scalability, and runs a limited set of options that are purchased separately
- Developer Edition – Has limited scalability and includes many of the options that are included in the Enterprise Edition.

For complete details on how to unlock editions and optional features, see the *Sybase® Software Asset Management Users Guide*.

Determining the Current Edition

Learn how to discover the edition you are currently running.

Enter:

```
sp_lmconfig 'edition'
```

Adaptive Server returns a value of EE, SE, or DE based on the edition that is currently running. See the *Reference Manual* for more information about the **sp_lmconfig** command.

Options for Adaptive Server Editions

Options that are available.

Table 1. Optional packages

Option	Platform availability	Description
Security and directory services	<ul style="list-style-type: none"> • HP-UX 64 • IBM AIX 64 • Linux 64 • Solaris 64 	Provides lightweight directory services and network-based authentication and encryption using SSL and Kerberos.
Partitions	<ul style="list-style-type: none"> • HP-UX 64 • IBM AIX 64 • Linux 64 • Solaris 64 	Enables semantic partitioning for table row data.
Encrypted Columns	<ul style="list-style-type: none"> • HP-UX 64 • IBM AIX 64 • Linux 64 • Solaris 64 	Increases security parameters and allows for addition of datatypes.
Tivoli Storage Manager	<ul style="list-style-type: none"> • HP-UX 64 • IBM AIX 64 • Linux 64 • Solaris 64 	Enables the database to backup and restore operations to IBM Tivoli Storage Manager.
In Memory Database	<ul style="list-style-type: none"> • HP-UX 64 • IBM AIX 64 • Linux 64 • Solaris 64 	Provides zero-disk footprint in-memory database support fully integrated with the Adaptive Server Enterprise for high-performance transactional applications. Provides performance enhancements to disk-resident databases with relaxed durability properties.

Adaptive Server editions and optional features are unlocked by SySAM licenses. For more information, see the *Sybase Software Asset Management Users Guide*.

Installation Workflow

Learn about installation prerequisites and the installation workflow.

1. Install operating system patches, and verify that the operating system meets the version-level, RAM, and network protocol requirements specified in System Requirements.
2. Before installing the software, shut down earlier versions of Adaptive Server.
3. Create a “sybase” account on your system to perform all installation tasks.
Create all devices using this “sybase” account.
4. Maintain consistent ownership and privileges for all files and directories. A single user—the Sybase system administrator with read, write, and execute permissions—should perform all installation, upgrade, and setup tasks.
5. If you are using an account that is local to only one physical box, create the "sybase" user account on each machine with the same user ID.
6. Ping the host to verify that your network software is configured.
7. Adjust the operating system shared memory parameter for your platform. See [Adjusting the Operating System Shared Memory Parameter](#) for information on how to do this.
8. Obtain your host ID. See [Obtaining The Host ID](#).
9. Decide which SySAM model to use. See [Obtaining a License](#).
10. Access SPDC. See [Accessing SPDC](#).
11. Generate a license. See [Generating a License](#).

Style Conventions

Style conventions used.

- In a sample window display, commands to be entered exactly as shown appear in:
`this font`
- In a sample window 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, 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 2. SQL syntax conventions

Key	Definition
command	Command names, command option names, utility names, utility flags, and other keywords are in a bold san-serif font.
<i>variable</i>	Variables, or words that stand for values that you fill in, are in <i>italic</i> .
{ }	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.
()	() 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, which are typed as part of the command.

Server Component Descriptions and Directory Layout

Learn about the server components and directory layouts.

Table 3. Product Description

Product	Description
Adaptive Server	<p>Installed into the ASE-15_0 directory.</p> <ul style="list-style-type: none"> Adaptive Server – the database server. Backup Server – an Open Server™-based application that manages all database backup (dump) and restore (load) operations. Monitor Server – an Open Server-based application that obtains performance information on Adaptive Server and makes that information available to Monitor Server client applications. XP Server – an Open Server application that manages and executes extended stored procedures (ESPs) from within Adaptive Server. Job Scheduler – provides a job scheduler for Adaptive Server. Job Scheduler components are located in their own directory at ASE-15_0/jobscheduler/.

Product	Description
Software Developer Kit (SDK)	<p>Installed into the <code>OCS-15_0</code> directory.</p> <ul style="list-style-type: none"> • Open Client (<i>Client Library, dclib</i>) • ESQ/C • Monitor Client Library • ESQ/COBOL • XA • ODBC (Windows, Solaris SPARC, Solaris x64, Linux Power, HP-UX Itanium, IBM AIX) – a driver used to connect to Adaptive Server from ODBC based applications. • OLEDB (Windows only) – a provider used to connect to Adaptive Server from OLEDB-based applications. • ADO.NET (Windows only) – a provider used to connect to Adaptive Server from .NET based applications.
shared	<p>The <code>shared</code> directory contains components and libraries that are shared by several other components. In earlier versions, this directory was named <code>shared-1_0</code>.</p> <ul style="list-style-type: none"> • Sybase Central 6 – is a Java-based framework used by the system management tools. • JRE – the Java Runtime Environment (JRE) is a runtime Java virtual machine used to execute Java-based programs such as Sybase Central. Adaptive Server includes the JRE. In typical installations, by default, the JRE is installed in full. In custom installations, if you select a component for installation that requires the JRE, the JRE is also automatically selected for installation. It is located in the <code>shared/JRE-6_0 * </code> directory
Adaptive Server Plug-in	<p>Installed into <code>ASEP</code> directory. This plug-in contains the management software for managing Sybase servers and its associated servers.</p>
Language Modules	<p>Installed into the <code>locales</code> directory. Provides system messages and date/time formats.</p>
Character Sets	<p>Installed into the <code>charsets</code> directory. Provides character sets available for use with Adaptive Server.</p>
Collation Sequences	<p>Installed into the <code>collate</code> directory. Provides the collation sequences available for use with Adaptive Server.</p>
Sybase Software Asset Management (SySAM)	<p>Installed into the <code>SYSAM-2_0</code> directory. Provides asset management of Adaptive Servers and optional features.</p>

Product	Description
Web Services	Installed into the <code>WS-15_0</code> directory. An optional product that provides access to Adaptive Server using HTTP/SOAP and WSDL.
Agent Management Console	This is a Sybase Central plug-in that manages the Unified Agent. It is installed into the <code>AMCP</code> directory.
Interactive SQL	Interactive SQL is a GUI tool that allows you to execute SQL statements, build scripts, and display data from Adaptive Server. It is installed into the <code>DBISQL</code> directory.
Unified Agent	Unified Agent is a management framework that provides runtime services to manage, monitor, and control distributed Sybase resources. Agents plug into the framework and provide the ability to manage specific components. It is installed into the new <code>UA</code> directory <code>UAF-2_0</code> directory.
jutils-3_0	A collection of Adaptive Server utility programs including: <ul style="list-style-type: none"> • ribo – a tool to trace TDS traffic between client program and Adaptive Server.
jConnect	Installed into the <code>jConnect-6_0_5</code> and <code>jConnect-7_0</code> directories. Provides a Java Database Connectivity (JDBC) driver for Adaptive Server.

PC-Client Product Descriptions and Directory Layout

Learn about the PC-Client product and directory layout.

Table 4. Products and Platforms

Product	Description
Software Developer Kit (SDK)	<p>Installed into the <code>OCS-15_0</code> directory.</p> <ul style="list-style-type: none"> • Open Client (<i>Client Library</i>, <code>dblib</code>) • ESQ/C • Monitor Client Library • ESQ/COBOL • XA <p>Installed into the <code>DataAccess</code> directory:</p> <ul style="list-style-type: none"> • ODBC (Windows and Linux only) – a driver used to connect to Adaptive Server from ODBC based applications. • OLEDB (Windows only) – a provider used to connect to Adaptive Server from OLE DB-based applications. • ADO.NET (Windows only) – a provider used to connect to Adaptive Server from .NET based applications.
Shared	<p>The shared directory contains components and libraries that are shared by several other components. In earlier versions, this directory was named <code>Shared</code>.</p> <ul style="list-style-type: none"> • Sybase Central 6 – Sybase Central is a Java-based framework used by the system management tools. • JRE – The Java Runtime Environment (JRE) is a runtime Java virtual machine used to execute Java-based programs such as Sybase Central.
Adaptive Server Plug-in	<p>Installed into <code>ASEP</code> directory. This plug-in contains the management software for managing Adaptive Server and its associated servers.</p>
Language Modules	<p>Installed into the <code>locales</code> directory. Provides system messages and date/time formats.</p>
Character Sets	<p>Installed into the <code>charsets</code> directory. Provides character sets available for use with Adaptive Server.</p>
jConnect	<p>Installed into the <code>jConnect-6_0_5</code> and <code>jConnect-7_0</code> directories. Provides a Java Database Connectivity (JDBC) driver for Adaptive Server.</p>

Product	Description
Agent Management Console	This is a Sybase Central plug-in that manages the Unified Agent. It is installed into the AMCP directory.
Interactive SQL	Interactive SQL is a GUI tool that allows you to execute SQL statements, build scripts, and display data from Adaptive Server. It is installed into the DBISQL directory.
jutils-3_0	A collection of Adaptive Server utility programs including: <ul style="list-style-type: none"> • ribo – a tool to trace TDS traffic between client program and Adaptive Server.

Separately Installable PC-Client Products

The PC-Client CD includes several additional products that install separately.

- InfoMaker® – a tool for personal data access, management, and reporting, used by both developers and end users. InfoMaker allows you to create InfoMaker presentation-quality reports and powerful queries. It complements client/server business applications and development tools as well as desktop productivity suites.
- PowerDesigner® Physical Architect – a tool for data modeling, including database design, generation, maintenance, reverse engineering, and documentation for database architects.
- PowerTransfer – a plug-in for PowerDesigner that transfers table data from any supported database to Adaptive Server.

Planning the Installation

Plan for the installation procedure.

Note: There is a change to the license quantity used by per CPU and per chip license types. Adaptive Server 15.5 checks out the same number of licenses as the number of cores on the machine (or chips, depending on the license type) regardless of any configuration settings. This behavior is not a change in licensing terms, but instead is a correction from earlier versions of the server, in which if Adaptive Server was licensed per CPU or CPU chip, therequested license quantity was reduced if the **max online engines** configuration parameter was set to fewer than the number of CPUs on the machine.

1. Determine the license type and the host ID.

Note: If you are configuring Adaptive Server Enterprise Active Messaging Version 15.5, you must have the correct license during the installation

2. Determine where to install the server.
3. Ensure your network connection is working.

Note: If you are using a SySAM License Server Adaptive Server 15.5 is linked with the FLEXnet Publisher version 11.5. If you use a SySAM License Server, you must update the license server to at least this version before installing the new server. To check the version information of your license server, examine the license server log or execute the following command:

```
cd $SYBASE/SYSAM-2_0/bin
./lmutil lmver lmgrd
```

Refer to Installing a new license server section for instructions on how to install the license server. If you are managing FLEXnet licenses from multiple vendors, refer to SySAM Users Guide for more information.

Obtaining a License

For simple installations, follow these instructions to obtain and install a SySAM license.

Prerequisites

- Before you install your product, obtain your license files from the *Sybase Product Download Center (SPDC)* before you install.
- Decide which SySAM license model to use:

Planning the Installation

- The unserved license model – obtains licenses directly from the license file. To use an unserved license, save the license to the same machine on which you are installing the Sybase product.
- The served license model – uses a license server to manage the license allocation to multiple machines.

To use a served license, you must have a license server. You can run the license server on the same machine or a different machine with a different operating system and architecture.

To use an existing license server, you must know the server host name and port number.

Note: There can be only one instance of a SySAM license server running on a given machine. To setup a SySAM 2.0 license server on a machine that is already running a SySAM 1.0 license server, you must migrate the old license server to SySAM 2. A migrated license server can serve licenses for both SySAM 1.0 and SySAM 2-enabled products.

Refer to the *Sybase Software Asset Management Users Guide* for more information about:

- Served and unserved licenses, and to decide which license model is best suited for your environment.
- Choosing a license server host.
- License server administration.
- Migrating a license server.
- If your installation is complex, for example it uses three-server redundancy, high availability, or virtualization, Sybase recommends that you see the *Sybase Software Asset Management Users Guide* about correctly setting up your license model.

Accessing SPDC

Once you have purchased a Sybase product, download the product and generate the licenses from the online Web portal, the Sybase Product Download Center (SPDC.)

1. After you order a Sybase product, you receive an email message that contains a URL to the SPDC site, as well as a user name and password.

Note: If you have purchased your Sybase software from a Sybase reseller, you will receive a Web key rather than an e-mail message.

2. Click on the URL, and when prompted, enter the designated user name and password. Your login is typically your e-mail address. If you have forgotten your password, use the password finder on the SPDC login page.

Next

Sybase Product Download Center

Generating a Served License

Before you install your Sybase product set up the license server and required licenses.

Prerequisites

Before generating the licenses, determine the host ID. See *Obtaining the SySAM Host ID* on page 28.

- If you have problems, see the Troubleshooting SySAM topic.

Task

1. From the welcome screen on the SPDC site, select the product family that contains the product for the license you want to generate.
2. From the list of Sybase products select the appropriate product edition, and version. Make sure you select the product for the operating system you are using.
The license agreement displays.
3. If the license agreement screen displays, select the country or region to display the applicable license text, read the agreement, then click **I Agree**.

Note: The license agreement appears only the first time you attempt to download a product. Once agree to the license, you do not see the license agreement for all subsequent downloads of that product.

4. On the product download page, click **License Keys**.
The License Information page displays.
5. Select the license to generate.

Verify that you are generating the correct license by comparing the product name, edition, and license type with your company's purchase order.

Note: If your product requires both a server and a client license, Sybase recommends that you generate the server license first.

6. Click **Select to Generate**.
7. Select **Served License**, then click **Next**.
8. Enter the quantity of licenses to generate, then click **Next**. You can generate licenses for a maximum of ten machines.
9. Enter the License Server Host ID and, optionally, the host name and port number.
 - a) A port number is not required unless you are using a three-node redundant cluster. A valid number is any unused port number between 0 and 64000. On UNIX, choose a port greater than 1024, since those less than 1024 are privileged port numbers. If you do not specify a TCP/IP port number, one of the default ports in the range of 27000 and 27009 is used.

Planning the Installation

- b) To generate a license for a three-node redundant cluster, specify the host ID, host name, and port number for each node of the cluster. In this configuration the port number is required and should be outside of the 27000 to 27009 range.
- c) Specify redundant server information to configure your servers for three-server redundancy.

10. Click **Generate**.

11. If you have just generated a server license your product also requires a client license, repeat steps 5-10 above for the client license.

12. Once you have generated all the licenses your product requires, you can:

- Download the license file by clicking the **Download All Licenses for Host** button.
- Print a paper copy of the license by clicking the **Print Friendly** button.
- Return to the license information page to generate additional licenses.

13. After generating the licenses, save them with a `.lic` extension to the `licenses` directory in your license server installation.

Generating an Unserved License

Learn how to generate an unserved license.

Prerequisites

Before generating licenses, determine the host ID. See *Obtaining the SySAM Host ID* on page 28.

- If you have problems, see the Troubleshooting topic.

Task

1. From the welcome screen on the SPDC site, select the product family that contains the product for the license you want to generate.
2. From the list of Sybase products select the appropriate product edition, and version. Make sure you select the product for the operating system you are using. The license agreement displays.
3. If the license agreement screen displays, select the country or region to display the applicable license text, read the agreement, then click **I Agree**.

Note: The license agreement appears only the first time you attempt to download a product. Once agree to the license, you do not see the license agreement for all subsequent downloads of that product.

4. On the product download page, click **License Keys**. The License Information page displays.
5. Select the license to generate.

Verify that you are generating the correct license by comparing the product name, edition, and license type with your company's purchase order.

Note: If your product requires both a server and a client license, Sybase recommends that you generate the server license first.

6. Click **Select to Generate**.
7. Select **Unserviced License** and click **Next**.
8. Select the number of machines to license and click **Next**. You can generate licenses for a maximum of ten machines.
9. Enter the host ID, and optionally, the host name for each machine for which you want to generate a license.

For some license types, you must enter the number of CPUs or licenses for the machine where this license is used.

10. Click **Generate**.
11. If you have just generated a server license your product also requires a client license, repeat steps 5-10 above for the client license.
12. Once you have generated all the licenses your product requires, you can:
 - Download the license file by clicking the **Download All Licenses for Host** button.
 - Print a paper copy of the license by clicking the **Print Friendly** button.
 - Return to the license information page to generate additional licenses.
13. After generating the licenses, save them with a `.lic` extension to the `licenses` directory in your license server installation.

Installing a New License Server

Before installing a new license server ensure that your system is updated with the correct operating system patches for your platform.

1. To install the SySAM Suite using the GUI tool, launch the executable file and follow these steps.
2. To install the SySAM utility and license server separately follow these substeps:

- a) Add a Java Runtime Environment 1.6 to `$PATH`.

For C Shell enter:

```
setenv PATH <JRE 1.6>/bin:$PATH
```

For Bourne Shell enter:

```
set PATH=<JRE 1.6>/bin:$PATH; export PATH
```

- b) Launch the SySAM Utility installer program from command prompt in the CD drive. Enter:

```
./archives/sysam_util/setup.bin
```

- c) Launch the SySAM License Server installer program from command prompt in the CD drive. Enter:

```
./archives/sysam_server/setup.bin
```

Planning the Installation

3. The Welcome Window displays. Click **Next**.
4. Select the most appropriate country from the drop-down list.
5. Accept the license agreement.
6. Enter or select the destination directory.
7. Select **Typical Installation Set**.
8. Click **Install** to proceed with the installation.
9. View the panel that reports installation success or failure.
10. Start the license server.

Note: The license server cannot be started until there is at least one valid served license installed in the `licenses` directory located at `$SYBASE/SYSAM-2_0/licenses`.

11. If you plan to use sub-capacity licensing see the *SySAM Users Guide* for information on how to configure the `sysamcap` utility.
12. Obtain the host ID for the machine on which you run the license server. For information on how to generate the host ID, see *Getting your host ID*.
13. Go to the *Sybase Product Download Center (SPDC)*.
14. Generate the licenses for the products you want to install. For instructions on generating your licenses, see the *Generating licenses at SPDC* topic.
- 15.

From command prompt, navigate to installed directory to source the environment variables.

- C Shell

```
source SYBASE.csh
```

- Bourne Shell

```
./SYBASE.sh
```

16. Copy the license file in the `$SYBASE/SYSAM-2_0/licenses` directory on the network license server machine.
17. Refresh or restart the license server:

1. Change to:

```
$SYBASE/SYSAM-2_0/bin
```

2. If the license server is not already started, start it, by entering:

```
sysam start
```

If the license server is already started, use the `reread` command to make the license server read the new license files:

```
sysam reread
```

18. Refresh or restart the license server:
19. Validate that the license daemon is running by entering:

```
sysam status
```

You can expect to see output for a running server that is similar to:

```
> sysam start
```

```
Starting the SYSAM service
The SYSAM service is starting.
The SYSAM service was started successfully.
```

20. Validate that the license daemon is running by entering:

```
sysam status
```

You can expect to see output for a running server that is similar to:

```
Starting the SYSAM service
The SYSAM service is starting.
The SYSAM service was started successfully.
```

```
lmutil - Copyright (c) 1989-2006 Macrovision Europe
Ltd. and/or Macrovision Corporation. All Rights
Reserved. Flexible License Manager status on Wed
5/24/2006 15:04
```

```
License server status: 27000@mysysamserver
License file(s) on keyserver:
/opt/sybase/SYSAM_0/licenses/mysysamserver_603020315.lic
```

21. The SySAM License Server installer installs both IPv4 and IPv4/IPv6 dual stack License Server binaries during installation and configures use of IPv4 only binaries. You can use the SySAM script to configure the appropriate version of the license server:

```
sysam configure [IPv6|IPv4]
```

For example, this command configures use of the IPv4/IPv6 dual stack binaries:

```
sysam configure IPv6
```

22. The IPv4 version of the license server cannot be used properly if you have enabled IPv6 on the license server host. If this is the case, you can:

- Disable IPv6 TCP/IP protocol on the license server host.
- Use another Windows host that does not have IPv6 enabled.
- Use a UNIX host for the license server. License Servers on UNIX are able to serve licenses to both IPv4 and IPv6 client hosts.
- If you must use a Windows host for the License Server and have both IPv4 and IPv6 clients on the license server, setup two License Servers, one serving the IPv4 network and the other serving the IPv6 network.

If you are using the IPv6 version of the license server on a Windows host, then this license server can be accessed only through IPv6 protocol. Any host with an IPv4 only network stack will not be able to obtain licenses from this IPv6 license server.

If you start the SySAM server from Windows services in Windows 7.0, Vista, or Windows 2008, the server starts successfully without any extra steps.

System Requirements for Servers

Before installing Adaptive Server ensure that your system is updated with the latest patches and system requirements. Do not use a patch that is earlier than the version suggested for your operating system. Use the patch recommended by the operating system vendor, even if it supersedes the patch listed.

To list all currently installed patches and display the operating system version level, enter:

- For Linux:

```
rpm -q -a
```

Note: The Java Virtual Machine (JVM) and associated Adaptive Server support requires a minimum of 250MB of virtual memory space. You may need more memory depending on the requirements of the individual java executions. You may need to adjust the memory parameters when running java to ensure that the java has enough virtual memory space to start, and that both Adaptive Server and java tasks successfully coexist.

In particular, systems that have Adaptive Server total memory set to a value higher than 1.5GB (or 2.5GB on some Enterprise servers) may experience problems and may have to reduce Adaptive Server total memory.

Table 5. System Requirements for Linux X86 and X64

Operating system	Hardware	Minimum RAM
Linux x86 Red Hat Enterprise Linux 4.0 update: 5	Intel Xeon AMD Opteron	512 MB
<ul style="list-style-type: none"> • kernel-2.6.9-55.EL • glibc-2.3.4-2.36 		
Red Hat Enterprise Linux 5.0 update:1		
<ul style="list-style-type: none"> • kernel-2.6.18-53.1.13.el5 • glibc-2.5-18 		
SuSE Linux Enterprise Server SLES 10		
<ul style="list-style-type: none"> • kernel-smp-2.6.16.21-0.8 • glibc-2.4-31.2 		

System Requirements for Servers

Operating system	Hardware	Minimum RAM
<p>Linux x64 RedHat Enterprise Linux 4.0 update: 5</p> <ul style="list-style-type: none"> kernel-2.6.9-55.EL glibc-2.3.4-2.36 <p>Red Hat Enterprise Linux 5.0 update:1</p> <ul style="list-style-type: none"> kernel-2.6.18-53.1.13.el5 glibc-2.5-18 <p>SuSE Linux Enterprise Server SLES 10</p> <ul style="list-style-type: none"> kernel-smp-2.6.16.21-0.8 glibc-2.4-31.2 	<p>Intel Xeon EM64T AMD Opteron</p>	<p>512 MB</p>
<p>Linux pSeries Red Hat Enterprise Linux 4.0 update: 5</p> <ul style="list-style-type: none"> kernel-2.6.9-55.EL glibc-2.3.4-2.36 compat-libstdc++-33-3.2.3-47.3 <p>Red Hat Enterprise Linux 5.0 update:1</p> <ul style="list-style-type: none"> kernel-2.6.18-53.1.13.el5 glibc-2.5-18 <p>SuSE Linux Enterprise Server SLES 10</p> <ul style="list-style-type: none"> kernel-smp-2.6.16.21-0.8 glibc-2.4-31.2 	<p>IBM pSeries Server with POWER5 CPU</p>	<p>512 MB</p>
Memory needed for 32-bit x86 systems		
Minimum RAM required for Adaptive Server	96MB	With default stack size, packet size, and user log cache size
Minimum RAM per additional user	Approx. 182KB	
Default user stack size	64KB	
Memory needed for 64-bit x64 systems		

Operating system	Hardware	Minimum RAM
Minimum RAM required for Adaptive Server	106MB	With default stack size, packet size, and user log cache size
Minimum RAM per additional user	Approx. 233KB	
Default user stack size	86K	
Memory needed for pSeries systems		
Minimum RAM required for Adaptive Server	127MB	With default stack size, packet size, and user log cache size
Minimum RAM per additional user	Approx. 324KB	
Default user stack size	128K	

Table 6. Disk Space Requirements for Linux X64 Platforms

Product	Disk space requirements
Adaptive Server typical install	921MB
Default databases created during install	306MB
Total	1071MB

Table 7. Disk Space Requirements for Linux X86 Platforms

Product	Disk space requirements
Adaptive Server typical install	865MB
Default databases created during install	306MB
Total	1015MB

Table 8. Disk Space Requirements for Linux X64 PSeries Platforms

Product	Disk space requirements
Adaptive Server typical install	1092MB
Default databases created during install	164MB
Total	1256MB

Note: Adaptive Server 15.5 supports Pentium and higher chip sets.

Sybase Support for OLE DB and ODBC

Adaptive Server 15.5 includes new ODBC and OLE DB drivers developed by Sybase. The third-party rebranded ODBC and OLE DB Driver Kits included with earlier Adaptive Server versions are no longer shipped.

The retired ODBC Driver Kit was installed in %SYBASE%\ODBC, and registered with the ODBC Driver Manager as Sybase ASE ODBC Driver. The Adaptive Server 32-bit ODBC Driver by Sybase is installed in %SYBASE%\DataAccess\ODBC, the 64-bit ODBC driver is at %SYBASE%\DataAccess64\ODBC and registered as Adaptive Server Enterprise. The version shipped with Adaptive Server 15.5 is version 15.0.0.1000.

The retired OLE DB Driver Kit was installed in %SYBASE%\OLEDB, and used the provider short name of “Sybase.ASEOLEDBProvider” and the long name of “Sybase ASE OLE DB Provider”. The new ASE OLE DB Provider by Sybase is installed in %SYBASE%\DataAccess\OLEDB, and %SYBASE%\DataAccess64\OLEDB uses the provider short name of ASEOLEDB. The version shipping with Adaptive Server 15.5 is version 15.0.0.1000.

Sybase recommends that you start your migration process to the new ODBC and OLE DB drivers as soon as possible. New features introduced in Adaptive Server 15.5 are supported only by the new drivers.

See “New Features for Open Server 15.5 and SDK 15.5 for Windows, Linux and UNIX,” for instructions on migrating to the new drivers.

Memory Requirements

Each operating system has a default maximum shared-memory segment. Make sure your operating system is configured to allow the allocation of a shared-memory segment at least as large as the max memory. Look here to determine the amount of memory Adaptive Server can use on your platform.

Platform	32-bit	64-bit
HP-UX 11.x (PA-RISC processor)	2.75 gigabytes	16 EB1
IBM AIX	5.x 2.75 gigabytes	16 EB
Sun Solaris 8 (SPARC processor)	3.78 gigabytes	16 EB
Sun Solaris 8 (Intel x86 processor)	3.75 gigabytes	chip dependent

Platform	32-bit	64-bit
Red Hat Enterprise Linux (Intel x86 processor)	2.7 gigabytes	chip dependent

System Requirements for Clients

Learn about the system requirements for PC-Client.

Table 9. Operating system requirements for the PC-Client

Product	Hardware	Operating system	Updates	Suggested minimum RAM
PC-Client	P4 1.0GHz	WinXP Pro SP2, Win2003 Server SP1, WinVista Business Edition		512MB

Note: If you are using ODBC, OLE DB or ADO.NET Drivers, then you must have Microsoft .NET Framework 2.0 Service Pack 1 installed on your system. The installer produces errors during installation and the drivers do not function without this service pack installed.

To check whether you have this service pack installed:

- Open Add or Remove Programs item from the Control Panel. If .NET Framework is listed under the Currently installed programs then you do not need to update your system.

System Requirements for Clients

Preinstallation Tasks for the Server Installation

Learn what to do before you begin the server installation.

1. Verify that the operating system meets the version-level, RAM, and network protocol requirements for your platform.
2. Read the release bulletins for the latest information on the products (Adaptive Server, Monitor Server, and so on) that you are installing. Check for any special installation instructions in the “Special Installation Instructions” section of the release bulletin.
3. Install operating system patches, if required.
4. Ensure that the account used to start the server has read and write permissions over the directory in which you are installing the server. You may need to create non-administrative logins to install the server into personal directories.
5. If you are installing on a Linux system and unset the LD_ASSUME_KERNEL variable before installing the server.
6. Ensure you have the required operating system patches for the Java Runtime Environment (JRE) version 6.

Information about the required operating system patches is available from the *IBM Java Web site*.

7. Review the SySAM procedures and plan your server configuration using the configuration guide for your platform. If you are installing the Cluster Edition, configuration information is in the installation guide for your platform. See the *Cluster Users Guide* for detailed information on configuration options.
8. Create a “sybase” account on your system to perform all installation tasks.

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

9. Log in to the machine as the “sybase” user.

Maintain consistent ownership and privileges for all files and directories. A single user—the Sybase system administrator with read, write, and execute permissions—should perform all installation, upgrade, and setup tasks.

10. If you are installing Adaptive Server 15.5 over an existing Adaptive Server 15.0.x version, and the Java in the database feature is enabled, perform these steps:
 - a) Create the **sybpcidb** database, which stores configuration information for the Pluggable Component Interface (PCI) and all Pluggable Component Adapter (PCA) components. This database is used by the **installpcidb** script. For example:

```
1> disk init
2> name = "sybpcidb_dev",
3> physname = "${SYBASE}/data/sybpcidb_dev.dat",
```

Preinstallation Tasks for the Server Installation

```
4>size = '24MB'  
5>go  
6> create database sybpcidb on sybpcidb_dev = 24  
7> go
```

- The device and database sizes depend on the Adaptive Server page size:
 - 2k page size – 24MB
 - 4k page size – 48MB
 - 8k page size – 96MB
 - 16k page size – 192MB
- b) If you are upgrading from a 15.0.x Adaptive Server, disable the Java feature before you upgrade, and reenale it after you have finished the upgrade. To disable the Java feature, use:

```
1> sp_configure 'enable java',0  
2> go
```

11. Decide where to install the software. Ensure there is sufficient available disk space. There cannot be any spaces in the path name of the directory.
12. Know the product edition you are installing. The options are:
 - Enterprise Edition — no license type needed
 - Small Business Edition — license type needed
 - Developers Edition — license type needed

For more information about license types, see the Getting and Using your License section of the *Sybase Software Asset Management Users Guide*.

Note: Sybase recommends that you specify license types at the time of installation. However, you may choose to specify the product edition and license type later using **sp_lmconfig**. For information on **sp_lmconfig**, see *Reference Manual:Procedures*.

13. You must decide if you want licensing events to trigger e-mail alerts and the severity of the events that generate e-mail messages.

If you choose to have e-mail notifications for license events, you must know the:

- SMTP server host name
- Port number for an SMTP server
- E-mail return address
- Recipients of the notifications
- Severity level of an event that triggers mail. Your choices are:
 - None
 - Informational
 - Warning
 - Error

14. Verify that your network software is configured.

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

If you are having connection problems, or to verify your network configuration, ping the host.

15. To optimize asynchronous I/O and to prevent the paging of shared memory, you must grant MLOCK permissions before you install Adaptive Server. To grant MLOCK permissions to a group, enter:

```
/etc/privgrp add mlock <group name>
```

or:

```
/etc/setprivgrp <group name> MLOCK
```

16. The operating system shared memory default, for most Linux releases is 32MB. The minimum required by Adaptive Server is 64MB for default Server with 2K pages. A higher value is required if you plan to increase Adaptive Server's **total memory**. Set the stack size to **unlimited** to avoid shared memory attachment problems. Set this using **ulimit -s unlimited** on Bourne Shell, or **limit stacksize unlimited** on C-Shell.
17. Use the **sysctl(8)** method to check and adjust the operating system shared memory parameter:

To check the current shared memory size, enter:

```
# /sbin/sysctl kernel.shmmax
```

To adjust the shared memory size at runtime, enter the following, where *nnn* is 1073741824 for 1GB of shared memory:

```
# /sbin/sysctl -w kernel.shmmax=nnn
```

Note: The default operating system shared memory for some older Linux releases is 32MB. The minimum required by Adaptive Server is 64MB for a default server with 2K pages. A higher value is required if you plan to increase the Adaptive Server **total memory**.

18. To guarantee that the minimum operating shared value is applied every time the system is started, edit the `/etc/sysctl.conf` file.
19. On Linux RHEL Update 4.0 and greater, Adaptive Server version 15.x running on multiple engines requires the security feature Exec-Shield to be disabled. To disable Exec-Shield:

- a) Add the following lines in `/etc/sysctl.conf` :

```
kernel.exec-shield=0
kernel.exec-shield-randomize=0
```

On RHEL Update 5.0

```
kernel.exec-shield=0
kernel.randomize_va_space=0
```

- b) As super user (root), for the action to take effect, enter:

```
/sbin/sysctl -P
```

For additional information, see *the Red Hat website*.

Obtaining the SySAM Host ID

Learn how to obtain your SySAM host ID.

1. Change to `SYBASE/SYSAM-2_0/bin`, Where `SYBASE` is the Sybase installation directory.
2. To obtain a host ID for a served license enter:

Note: On some platforms, the host ID is derived from the network adapter address. If your machine has multiple network adapters, `lmutil lmhostid` returns one host ID for each network adapter. The output may look similar to:

```
The FLEXlm host ID of this machine
is "0013023c8251 0015c507ea90"
Only use ONE from the list of hostids.
```

Choose one of these host IDs. Sybase recommends that you use the value associated with the primary wired Ethernet adapter. Do not use values associated with internal loopback adapters. If you cannot determine which host ID to use from the `lmutil lmhostid` output, use an appropriate native operating system command to get additional details. See the SPDC FAQ topic titled, "What's my Host ID."

3. Make note of the host ID so you can use it later at SPDC.
 - Your host ID is platform-specific, but may be similar to:
 - 00400516E525 on Linux platforms.
 - 170a3472 on Solaris platforms.
 - fb491d9 on AIX platforms.
 - ID_STRING= 972179e8-7b2b-11d7-bd55-4a651849415a on HP Itanium

Preinstallation for an Overlay Installation

Learn how to execute an overlay installation.

1. Verify that your current Adaptive Server is an earlier version. In `isql` enter:

```
1> select @@version
2> go
```
2. Installing a newer version of Adaptive Server overwrites any older versions of Adaptive Server. Prior to installing make sure that your databases are error free and back up your `SYBASE` directory, in case you must return to the original version of the software. To ensure that your database is error free, run **dbcc checkdb**, **dbcc checkcatalog** and **dbcc checkstorage** prior to installing new Adaptive Server binaries including the *master* database.

If any of the **dbcc** commands reveal problems, stop here and see the *Troubleshooting and Error Messages Guide* for more information.

3. If you are installing Adaptive Server 15.5 over an existing Adaptive Server 15.0.x version then, if Java in the database feature is enabled, perform the following steps:

- Create the **sybpcidb** database

The **sybpcidb** database stores configuration information for the PCI and all PCA components. This database is used by the **installpcidb** script. For example:

```
1> disk init
2> name = "sybpcidb_dev",
3> physname = "${SYBASE}/data/sybpcidb_dev.dat",
4> size = '24MB'
5> go
1> create database sybpcidb on sybpcidb_dev = 24
2> go
```

The device and database sizes depend on the Adaptive Server page size:

- 2k page size - 24MB
- 4k page size - 48MB
- 8k page size - 96MB
- 16k pagesize - 192MB

If you are upgrading from a 15.0.x ASE, disable the Java in the database feature before upgrading and enable again after the upgrade.

- Disable the Java in the database feature:

```
1> sp_configure 'enable java', 0
2> go
```

Note: If you plan to use **sqlupgrade** utility to upgrade from a pre-15.0.x Adaptive Server, do not disable the Java in the database feature. The **sqlupgrade** utility will automatically disable and re-enable the Java in the database feature.

Preinstallation Tasks for the Server Installation

Installing Server Components

Sybase recommends installing products while logging in as a "sybase" user.

Note: Adaptive Server Enterprise 15.5 includes new major versions of Adaptive Server and many of the supporting components. Installing Adaptive Server version 15.5 into the same directory with existing products should not impact the existing products. However, installing other products on top of Adaptive Server version 15.5 results in one or more products not working correctly.

Sybase strongly recommends that you install Adaptive Server version 15.5 be installed into its own directory if possible. Where this is not practical and other products must be installed into the same directory, Adaptive Server version 15.5 should be installed last.

You must source the environment variables needed for Adaptive Server products by running the `SYBASE.csh` or `or SYBASE.shscript` file after exiting the installer.

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

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

Installing in GUI mode

Shut down all programs before running the installer.

1. Insert the Adaptive Server CD or DVD in the appropriate drive, or download and extract the Adaptive Server install image from the Sybase Product Download Center (SPDC).
2. Enter:


```
mount -v "cdrfs" -r <device> /cdrom
```
3. Launch the installer. Enter:


```
/cdrom/setup.bin
```
4. Select the language to install.
5. The Welcome screen displays. Click **Next**.
6. Select a directory for the installation. Click **Next** to accept the default directory, or enter a new directory path.
 - a) If you specify a destination that contains a previous version of the server, the **Choose Update Installation** panel displays. In this panel, you choose the features to update.
7. If you are updating your pre-existing server, you will not see this step. Choose the type of installation to be performed:

Installing Server Components

- Typical
 - Full
 - Custom
8. If you select Custom, the Products and Features Selection window displays.
- Select the products to install by selecting the box next to the product name. Some features are dependent on other features; therefore, the installer prevents you from unselecting some items without first unselecting others.
9. If you are updating your pre-existing server, you will not see this step. Select the software license type for the Adaptive Server suite. The available types to be selected are different for different platforms.
- Install licensed copy of Adaptive Server Suite
 - Install Free Developer Edition of Adaptive Server Suite
 - Install Express Edition of Adaptive Server Suite
 - Evaluate Adaptive Server Suite which is valid for 30 days from the installation date
10. Choose the most appropriate region, read the license terms, and then click “I agree.” Click **Next**.
11. If you choose Free Developer Edition, Express Edition or Evaluate Adaptive Server Suite you will not see this step. If you chose to install a licensed installation you can:
- Specify the license file — You are prompted to enter a license key and set up the license server.
 - Use an existing license server
 - Proceed without specifying license information for the duration of 30 days from the installation date.

If you choose Free Developer Edition, Express Edition or Evaluate Adaptive Server Suite you will not see this step. If you choose **Specify the License File** and entered a served license key, you are prompted to set up the license server.

12. If you choose Free Developer Edition, Express Edition or Evaluate Adaptive Server Suite you will not see this step. Select the Adaptive Server product edition.

The available editions to be selected are different for different platforms.

- Enterprise Edition
 - Small Business Edition
 - Developer Edition
 - Unknown
- a) Depending on the product edition selected, you can also select the license type under which the product is licensed. If you are unsure about the license type, select **Unknown**.

If you choose **Unknown** for the Product Edition and then choose to use the **Optimize Adaptive Server** task later in the installation, it will fail.

13. If you choose Free Developer Edition, Express Edition or Evaluate Adaptive Server Suite you will not see this step. Configuring your server for e-mail notification enables designated users receive information about license management events requiring attention.

Provide the following information:

- SMTP server host name
- SMTP server port number
- E-mail Return Address
- Recipient e-mail addresses
- Message severity that triggers e-mail messages

A summary window appears. Click **Next**.

14. Verify the type of installation, and that you have enough disk space to complete the process. Click **Next**.
15. The installation Status window displays the result of the installation process.
16. Select **enable** or **disable** the remember password feature then click **Next**.
17. You are prompted to configure new servers depending upon the products you installed.
- Configure a new Adaptive Server
 - Configure a new Backup Server
For information on developing a backup and recovery plan and backing up and restoring user databases, see the *System Administration Guide*.
 - Configure a new Monitor Server
Monitor Server requires additional configuration steps after the installation is complete. See the *Monitor Server Users Guide*.
 - Configure a new XP Server
For information on using extended stored procedures in XP Server, see the *Transact-SQL® Users Guide*.
 - Configure Job Scheduler
For information on using Job Scheduler, see the *Job Scheduler Users Guide*.
 - Enable Self Management
 - Configure Web Services
Web Services Users Guide explains how to configure, use, and troubleshoot Web Services for Adaptive Server.
 - Configure an Unified Agent
Unified Agent and Agent Management Console describes the Unified Agent, which provides runtime services to manage, monitor and control distributed Sybase resources.

Select the servers you want to configure. If you do not want to configure the new servers at this point, unselect the item. Click **Next**.

18. The Custom Configure Input window prompts you for the custom Adaptive Server configuration information.

Installing Server Components

- Server name
- Port number
- Error log – name and location of the error log file.
- Application type – the choices are:
 - MIXED – both OLTP & DSS. This is the default.
 - OLTP – online transaction Processing generally consists of smaller, less complex transactions.
 - DSS – decision Support systems generally have less update activity with large complex queries.
- Page size:
 - 2KB
 - 4KB – default
 - 8KB
 - 16KB

Attention: The default page size selection has changed to 4KB in this version of Adaptive Server. The previous default was 2KB. If you plan to load database dumps from Adaptive Server whose page size is not 4KB, then change this page size to match the database dump.

- Master device – name of master device and the path where it should be located.
- Master device. The default master device sizes are:
 - 2KB page size – 30MB
 - 4KB page size – 60MB
 - 8KB page size – 120MB
 - 16KB page size – 240MB
- Master database . The default master database sizes are:
 - 2KB page size – 13MB
 - 4KB page size – 26MB
 - 8KB page size – 52MB
 - 16KB page size – 104MB
- System procedure device path
- System procedure device
- System procedure device and database default size is 150MB.
- System device — The default sizes are:
 - 2KB page size server – 3MB
 - 4KB page size server – 6MB
 - 8KB page size server – 12MB
 - 16KB page size server – 24MB
- System database size (in MB) The default sizes are:
 - 2KB page size server – 3MB

- 4KB page size server – 6MB
- 8KB page size server – 12MB
- 16KB page size server – 24MB
- Temporary database device path
- Temporary database device size:
 - 2KB page size server – 100MB
 - 4KB page size server – 100MB
 - 8KB page size server – 100MB
 - 16KB page size server – 100MB
- Temporary database size:
 - 2KB page size server – 100MB
 - 4KB page size server – 100MB
 - 8KB page size server – 100MB
 - 16KB page size server – 100MB
- Enable PCI in Adaptive Server
 - PCI device and PCI database size are the same
 - 2KB page size server – 24MB
 - 4KB page size server – 48MB
 - 8KB page size server – 96MB
 - 16KB page size server – 192MB
- Optimization configuration:
 - Available physical memory for Adaptive Server - round down 80% of the combined physical memory and OS shared memory.
 - Available CPU for Adaptive Server – round down 80% of the physical CPU. The minimum is 1.

Note: If the value specified is larger than the available resource for allocation to the server, then the optimize configuration may fail causing the server to not start.

Once you have customized the Adaptive Server configuration, select **Next** to record the input fields.

19. Enter the Backup Server information.

Once you have entered the appropriate information, click **Next** to record the information.

20. Enter the Monitor Server configuration information.

Once you have entered the appropriate information, click **Next** to record the information.

21. Enter the XP Server configuration information.

Once you have entered the appropriate information, click **Next** to record the information.

22. Enter Job Scheduler information:

- Agent name

Installing Server Components

- Port number
- Management device
- Management device size
- Management database size

Once you have entered the appropriate information, click **Next** to record the information.

23. Custom configure Self Management by entering the following information:

- Self Management user name
- Self Management password

Note: If you choose the default user name “sa,” you cannot enter the password. The “sa” default password is null.

24. Custom configure Web Service by entering the following information:

- Producer HTTP Port Number
- Producer Host Name
- Certificate Password
- Confirm Certificate Password
- Keystore Password
- Confirm Keystore Password
- Producer Log File
- Consumer Port Number
- Consumer Log File

25. Security Login Modules – choose the security login modules for the Unified Agent. You can choose:

- Simple Login Module

Note: If you double-click Simple Login Module, a dialog asks you to change the user name and password.

- Adaptive Server Login Module
- Unix Proxy Login Module

Select the **Enable** box to enable a login module.

26. The Configure New Servers Summary window displays a summary of the values that are used to configure the servers. These values are either the default or custom values. After verifying the information, click **Next**, and the installer proceeds with the server configuration.

27. The Configure Server Progress window displays the progress of the server configuration.

28. The Restart window displays only when required. It reminds you to either log out or restart your machine to complete the installation process.

29. The Installation Complete window is the last window.

If you encounter any errors, see the *Troubleshooting Guide*.

30. At the end of a successful installation click **Done**.

Adaptive Server and related products have been successfully installed, and minimally configured for use. See *Installing sample databases*, to begin experimenting with your servers, or see the *System Administration Guide at Sybase Product Manuals* for more advanced topics.

Installing From the Command Line

The steps for installing components in console mode are the same as those described in the *Installing in GUI Mode* topic, except that you run the installer from the command line using the `setup` command, and you enter text to select the installation options. Look here to learn how to install from a command line.

1. At the command line, enter:

```
setup.bin -i console
```

The installation program starts.

2. In cases where the installation program launches automatically, select **Cancel** to stop the GUI installation, then launch the setup program from a terminal or console.

The flow of the installation is identical to a regular GUI installation, except that the display is written to a terminal window and responses are entered using the keyboard.

Response File Change

Please note changes to the installation response file from previous releases.

This version of Adaptive Server uses a rearchitected installation program. Response files generated for previous versions of Adaptive Server (including and up to 15.0.3) are not compatible. There is no facility to migrate these response files to the new format. Follow the steps described to create required response files in the new format.

Creating a Response File

To perform a silent, or "unattended" installation, run `InstallAnywhere` and provide a response file that contains your preferred installation configuration.

To create a response file when installing in GUI or console mode, specify the `-r` command line argument. The `-r` argument records your responses to the installation wizard's prompts and creates a response file when the `Install anywhere Wizard` exits. The response file is a text file that you can edit to change any responses prior to using it in any subsequent installations.

To generate the response file during GUI installation, run:

```
setup.bin -r responseFileName
```

`-r responseFile` is optional. If specified, the installation session is recorded to response file.

Where *responseFileName* is the absolute path of the file name you choose for the response file, for example:

```
/tmp/responsefile.txt
```

Note: The directory path you specify must already exist.

Installing Server in Silent Mode

After running the installer in GUI mode and recording the responses into a response file, edit the file to customize the responses. Then use that response file to answer the installer when you run the installer in silent mode.

Prerequisites

Use

```
setup.bin -r responseFileName
```

to generate a installation response file during a console or GUI installation.

To install in silent mode, run:

```
setup.bin -f responseFileName -i silent  
-DAGREE_TO_SYBASE_LICENSE=true -DRUN_SILENT=true
```

where *responseFileName* is the absolute path of the file name containing the installation options you chose.

Note: You must agree to the Sybase License Agreement when installing in silent mode. You can either:

- Include the option

```
-DAGREE_TO_SYBASE_LICENSE=true -DRUN_SILENT=true
```

in the command line argument.

- Edit the response file to include the property

```
AGREE_TO_SYBASE_LICENSE=true  
RUN_SILENT=true
```

Except for the absence of the GUI screens, all actions of InstallAnywhere are the same, and the result of an installation in silent mode is exactly the same as one done in GUI mode with the same responses.

Installing Adaptive Server Using Binary Overlay

Read “Special Installation Instructions” in the release bulletin for the latest information that may affect specific 15.5 configurations.

1. If you are upgrading from a pre-15.5 server and the Java in the database feature is enabled perform these steps:

- a) Create the **sybpcidb** database. The **sybpcidb** database stores configuration information for the PCI and all PCA components. This database is used by the **installpcidb** script. For example:

```
1> disk init
2> name = "sybpcidb_dev",
3> physname = "${SYBASE}/data/sybpcidb_dev.dat",
4> size = '24MB'
5> go
1> create database sybpcidb on sybpcidb_dev = 24
2> go
```

- b) Disable java feature:

```
1> sp_configure 'enable java', 0
2> go
```

Note: If you are upgrading from a pre 15.0.x server, do not disable Java in the database feature. **sqlupgrade** automatically disables this feature before an upgrade and re-enables the feature after an upgrade.

2. After backing up your databases, shut down Adaptive Server and back up your SYBASE directory.
3. Installing Adaptive Server 15.5 overwrites current Adaptive Server software. Before installing 15.5, make sure that your databases are error-free as described in Backing up Adaptive Server and that your SYBASE directory is backed up.
4. Shutdown servers that are using the binaries from your SYBASE directory.
5. Use InstallShield to load the new software from the CD or DVD.

For UNIX installations, navigate to the CD or DVD drive and enter `./setup.bin`.

6. Install Adaptive Server 15.5 over the `$SYBASE` installation path.

Note: Once files are loaded into your `$SYBASE` directory, InstallShield asks if you want to configure the new installed server. Deselect configuring the new server, and click continue to finish the installation.

7. Restart Adaptive Server.

The shell command is:

```
$SYBASE/$SYBASE_ASE/install/startserver [-f RUN_server_name file]
```

Select [Sybase SQLserver_<servername>] and stop and start the services.

8. Run select **@@version**. The server should now be at version 15.5.
9. In Adaptive Server version 15.5, several changes have been made to the system stored procedures and many new error messages have been added. After performing a binary overlay you must perform the postinstallation tasks to make these changes available.

Refer to the Post-upgrade tasks before running `installmaster` or `instmsgs.ebf`.

Determining the Adaptive Server Version

Learn how to determine the version of Adaptive Server you are currently running.

Verify that your current Adaptive Server installation is at version 15.x by running the following query from **isql**:

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

If your server is not running, you can get a version string by running:

```
$$SYBASE/$SYBASE_ASE/bin/dataserver -v
```

Note: You must source environment variables before executing **dataserver -v**.

If this shows that the Adaptive Server is not at version 15.x, you must upgrade Adaptive Server. See [Upgrading Adaptive Server](#).

If your server is at a 15.x version level, you can begin installing Adaptive Server 15.5.

Backing up the Server

Adaptive Server provides a script for backing up *sysmessages* before upgrading to 15.5.

Use **uninstmsgs.ebf** to back up *sysmessages* before running **instmsgs.ebf**. Refer to the [Post-upgrade tasks](#), section.

Installing Adaptive Server version 15.5 overwrites the current Adaptive Server software. Before installing, verify that your databases are error free and that your \$SYBASE directory is backed up.

To ensure that your database is error free, run **dbcc checkdb**, **dbcc checkcatalog** and **dbcc checkstorage** before loading any new Adaptive Server binaries, including the master database. If the **dbcc** commands reveal problems, check the *Error Messages and Troubleshooting Guide* for actions necessary to fix the problem. If the error is not listed in the manual, call Sybase Technical Support.

After you have verified that your database is error free, back up your \$SYBASE directory in case you need to roll back to the original version of the software.

Upgrading an Adaptive Server Configured with High Availability

Learn how to upgrade a server configured with high availability.

Refer to *Using Sybase Failover in a High Availability System* for instructions on how to upgrade an Adaptive Server configured with high availability.

Monitoring and Diagnostic (MDA) Tables

The `installmontables` script for installing Monitoring and Diagnostic tables (MDA) uses the materialized parameter when defining the MDA proxy tables.

This results in improved performance for many queries using the MDA tables, making it no longer necessary to define a **loopback** remote server in your `sys.servers` table in order to use the MDA tables.

Sybase recommends that you reinstall the MDA tables using the `installmontables` script contained in this release.

Note: No changes are needed to run the `installmontables` script unless you plan to use the MDA tables to monitor remote servers.

New Monitoring and Diagnostic Tables

The `mon_role` privilege is necessary to retrieve information on the PCI Bridge and its associated slots.

- `monPCIBridge` – a 1-row MDA table containing ASE/PCI Bridge information.
- `monPCIEngine` – a multi-row table containing 1 row for each online engine.
- `monPCISlots` – this is a multi-row table containing 1 rows for each active slot running under the PCI Bridge.

Monitoring Remote Servers Using MDA Tables

Learn how to monitor remote servers with MDA tables.

Make these changes to use the MDA tables to monitor remote servers:

1. Update the `installmontables` script by changing all occurrences of “materialized” to “external procedure.”
2. Modify the **at** clause in the proxy table definitions in the `installmontables` script to add the name of the remote server followed by the string “...” before the MDA RPC name.
3. You can identify MDA RPC names by the “\$” (dollar sign) preceding the name. The remote server name must be the same as it appears in the `interfaces` file on the server where the MDA tables are installed.

For example, for the `monState` table definition, if your remote server name appearing in the `interfaces` file is `SALESPROD`, change:

```
at "$monState"
```

to:

```
at "SALESPROD...$monState"
```

Installing Server Components

If the remote server is not registered with the server on which you are installing the MDA tables, use the **sp_addserver** stored procedure to register it.

In order to reference a remote server during the installation, be sure to use the version of the `installmontables` script that came with the version of the Adaptive Server Enterprise installed at the remote server.

Use of the materialized directive has changed the way some error messages from the MDA tables are sent to the client application. If all configuration parameters required by an MDA table used in a query are not enabled, then the client application receives an error message.

Removing Adaptive Server

Only actions performed by the installer are removed during an uninstallation process. This means that files or registry entries created after installation are not removed by the uninstaller and must be removed by the customer after the uninstall is completed. Where possible, these actions are identified.

If Adaptive Server version 15.5 is installed on top of an existing Adaptive Server 12.5 or earlier, Adaptive Server version 12.5 or earlier creates a shortcut that launches the uninstaller, while Adaptive Server 15.5 registers the uninstaller with the **Add/Remove Programs** option on the control panel. Use the **Add/Remove Programs** option to remove Adaptive Server. This does not remove some of the shortcuts created by the Adaptive Server version 12.5 or earlier installer; you must manually remove these shortcuts.

Uninstalling

Learn how to uninstall the server.

Note: Uninstalling Adaptive Server removes Sybase Central .jar files which may be shared by other Sybase products installed in the same \$SYBASE directory. A “files not found” error message displays when attempting to start Sybase Central after the uninstal. After the uninstallation, you must re-install the other Sybase products to \$SYBASE and overwrite the old products.

1. To check if the server is running use:

```
ps
```

- a) Then if it is running use **isql** commands:
kill or **shutdown** to shutdown the server.

All servers must be shut down prior to running the uninstall program. See “Starting and Stopping Servers” in the *Configuration Guide*.

2. Launch the uninstall program by executing the following steps:

- a) Change to the SYBASE directory.
- b) Execute the uninstall program by entering:

```
$SYBASE/sybuninstall/ASESuite/uninstall
```

3. The Uninstall Welcome window displays. Click **Next**.
4. Choose either:
 - Complete Uninstall.
 - Uninstall Specific Features
 - a) Click **Next**.
5. If you select to uninstall specific features, a confirmation list of features displays. If any remaining products/components on the computer are dependent on features selected for uninstall, you will be notified. You can still proceed with the uninstall but dependent features will not function.
6. A window displays indicating that the uninstallation process is being performed. There is no progress bar.

Note: One or more pop-up displays may occur indicating that a file was modified after it was installed and requesting confirmation that it should be deleted.

7. When the uninstall process has completed a final window displays. Click **Done** to exit the uninstall program.

Removing an Existing Adaptive Server

Learn how to remove an existing Adaptive Server.

1. From \$SYBASE, enter:

```
rm servername.*
```

2. Change to the directory \$SYBASE/\$SYBASE_ASE/install and run the following commands:

```
rm RUN_servername.*
```

```
rm servername.*
```

3. Edit \$SYBASE/interfaces, to remove all references to the Adaptive Server.
4. If you used operating system files for database devices, remove those.

Installer Changes from Adaptive Server 12.5.4 to Adaptive Server 15.5

From Adaptive Server 12.5.4 to 15.5 a number of changes to the directory structures occurred.

Table 10. Directory Changes for Unix Platforms

Component	12.5.4 location	15.0.2 location	15.0.3 location	15.5 location
Adaptive Server	\$SYBASE/ ASE-12_5	\$SYBASE/ ASE-15_0	\$SYBASE/ ASE-15_0	\$SYBASE/ ASE-15_0
Shared directory	\$SYBASE/ shared	\$SYBASE/ shared	\$SYBASE/ shared	\$SYBASE/ shared
Sybase Central	\$SYBASE/ shared/syb- central43	\$SYBASE/ shared/syb- central43	\$SYBASE/ shared/syb- central600	\$SYBASE/ shared/ sybcen- tral600
JRE	\$SYBASE/ shared/ jre142	\$SYBASE/ shared/ jre142_*	\$SYBASE/ shared/ JRE-6_0*	\$SYBASE/ shared/ JRE-6_0*
Shared JAR file	\$SYBASE/ shared/lib			
locales	\$SYBASE/lo- cales	\$SYBASE/lo- cales	\$SYBASE/lo- cales and \$SYBASE/ ASE-15_0/ locales	\$SYBASE/ locales and \$SYBASE/ ASE-15_0/ locales
Connectivity	\$SYBASE/ OCS-12_5	\$SYBASE/ OCS-15_0	\$SYBASE/ OCS-15_0	\$SYBASE/ OCS-15_0
Web Service	\$SYBASE/ WS-12_5	\$SYBASE/ WS-15_0	\$SYBASE/ WS-15_0	\$SYBASE/ WS-15_0
Replicator	\$SYBASE/ RPL-12_5	\$SYBASE/ RPL-15_0	\$SYBASE/ RPL-15_0	
SySAM	\$SYBASE/SY- SAM-1_0	\$SYBASE/SY- SAM-2_0	\$SYBASE/SY- SAM-2_0	\$SYBASE/ SYSAM-2_0

Component	12.5.4 location	15.0.2 location	15.0.3 location	15.5 location
Job Scheduler	\$SYBASE/ JS-12_5	\$SYBASE/ ASE-15_0/ jobschedu- ler	\$SYBASE/ ASE-15_0/ jobschedu- ler	\$SYBASE/ ASE-15_0/ jobschedu- ler
Unified Agent		UAF-2_0	UAF-2_0	UAF-2_5

PC-Client Installation

The PC-Client installer contains client applications and utilities you can use to access the Adaptive Server to execute queries or manage the server. It also contains the Sybase Open Client/ Open Server™ Software Developers Kit, which you can use to develop applications that access the server, and ODBC, OLE DB, and ADO.NET clients.

Client applications, such as Sybase Central and Open Client, are used to access Adaptive Server.

Sybase Central allows you to connect to Adaptive Server via client connections. You can use Sybase Central to connect to remote servers, local servers (installed on the same machine), and multiple servers. See the Getting Started After Installing topic for information on configuring network connections.

The Sybase PC-Client CD includes the Software Developer's Kit (SDK) for Windows platforms, consisting of:

- Embedded SQL/C (ESQL/C)
- Embedded SQL/Cobol (ESQL/Cobol)
- Monitor Client Library
- XA Interface Library for ASE Distributed Transaction Manager
- Additional Connectivity Language Modules
- Open Client (CT-Library, DB-Library)
- Agent Management Console
- Microsoft Cluster Server Resource Type for ASE
- Interactive SQL
- Sybase Central
- SySAM Plug-in
- ASE ADO.NET Data Provider
- ASE OLE DB Provider by Sybase
- ASE ODBC Driver by Sybase
- ASE plug-in
- QPTune
- jConnect™ for JDBC
- jConnect 6.0.5 for JDBC
- jConnect 7.0 for JDBC
- SySAM License Utilities
- DBISQL
- SDC Administration Utility

System Requirements for PC-Client

Learn about the Windows system requirements for PC-Client here.

Operating system	Service pack level
Windows 2003	Service pack 2
Windows XP Professional	Service pack 2
Windows Server 2003 Edition x64	Service pack 2

Note: If you are using ODBC, OLE DB or ADO.NET Drivers, then you must have Microsoft .NET Framework 2.0 Service Pack 1 installed on your system. The installer produces errors during installation and the drivers do not function without this service pack installed.

To check whether you have this service pack installed:

- Open Add or Remove Programs item from the Control Panel. If NET Framework is listed under the Currently installed programs then you do not need to update your system.
-

Installing the Client

The PC-Client CD contains several components that are each packaged with their own installer. When you insert the PC-Client CD, a menu program launches automatically. The menu program presents the list of components that can be installed from the CD and allows you to install them one at a time. Before installing any products, review the `readme.txt` file, which contains a brief description of each of the products, lists dependencies, and contains any last-minute instructions or changes.

Warning! The PC-Client now contains 32-bit and 64 bit SDK. It installs 32-bit SDK on 32-bit OS; and 64-bit SDK on 64-bit OS.

1. Verify that your computer has sufficient disk space for each product.
2. If you are unloading components, log in using an account with administrator privileges.
3. Close any open applications or utilities to free memory and system resources.
4. Go to **Start > Run** to start the installation program, and enter the following (where *X* is the directory into which you have copied the PC-Client installation files:


```
X:\autorun.exe
```
5. Select Client Components 15.5.
6. When the installer starts, the Welcome window displays. Click **Next**.

7. When the Sybase license agreement displays, select a country, click **I Agree**, then click **Next**.
8. Enter the directory path and click **Next**.
 - a) If you specify a destination that contains a previous version of the server, the **Choose Update Installation** panel displays. In this panel, you choose the features to update.
9. If you are updating your pre-existing server, you will not see this step. Choose the type of installation to be performed.
 - Typical Install – installs the default components that most users need.
 - Full Install – installs every component on the CD.
 - Custom Install – allows you to select the components to install. Certain components are automatically installed if they are required to run other selected components.
10. If you select Custom Install, next specify the components to install.

Components installed in a typical installation are automatically selected. You may select or unselect components from this list.
11. Click **Next**.

The Summary window displays every component to be installed by the installer, the required disk space, and the available disk space.

If the target directory does not have enough free space, the available space appears in red. You must either click Previous to return to the previous window and change your selections, or Cancel to quit the installer.
12. Click Next. The installer displays a progress indicator as it installs components.

Creating a Response File

To perform a silent, or "unattended" installation, run InstallAnywhere and provide a response file that contains your preferred installation configuration.

To create a response file when installing in GUI or console mode, specify the **-r** command line argument. The **-r** argument records your responses to the installation wizard's prompts and creates a response file when the Install anywhere Wizard exits. The response file is a text file that you can edit to change any responses prior to using it in any subsequent installations.

To generate the response file during GUI installation, run:

```
setup.bin -r responseFileName
```

-r responseFile is optional. If specified, the installation session is recorded to response file. Where *responseFileName* is the absolute path of the file name you choose for the response file, for example:

```
/tmp/responsefile.txt
```

Note: The directory path you specify must already exist.

Installing the Client in Silent Mode

After running the installer in GUI mode and recording the responses into a response file, edit the file to customize the responses.

To install in silent mode, run:

```
setupConsole.exe -f responseFileName -i silent  
-DAGREE_TO_SYBASE_LICENSE=true
```

where *responseFileName* is the absolute path of the file name containing the installation options you chose.

Note: You must agree to the Sybase License Agreement when installing in silent mode. You can either:

- Include the option

```
-DAGREE_TO_SYBASE_LICENSE=true
```

in the command line argument.

- Edit the response file to include the property

```
AGREE_TO_SYBASE_LICENSE=true
```

Except for the absence of the GUI screens, all actions of `InstallAnywhere` are the same, and the result of an installation in silent mode is exactly the same as one done in GUI mode with the same responses.

Warning! Sybase recommends that you use the `setupConsole.exe` executable, which runs in the foreground when you are running a silent installation. The normal `setup.exe` executable runs in the background, giving you the impression that the installation has terminated, and results in additional installation attempts using the silent installation. Multiple installations at the same time may corrupt the Windows Registry and lead to a failure to restart the operating system.

Getting Started After Installing

After installing Adaptive Server, Sybase Central, and the Java Runtime Environment, test the installation and network connections. 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 must know where the others reside on the network. This information is stored in the `interfaces` file on Windows or in a Lightweight Directory Access Protocol (LDAP) server.

1. Select **Start > Programs > Sybase > Sybase Central 6.0.0**.
2. Select **Tools Connect** from the Sybase Central menu bar to activate an Adaptive Server login window.

3. Log in using the default user ID, “sa”, without a password. If you have changed the password, use the new password.
4. Select the Adaptive Server to which you want to connect.
5. To disconnect from a server, select **Tools** Disconnect, or right-click the server icon to which are connected and select Disconnect.
6. Exit Sybase Central.

Configuring libtcl.cfg for LDAP

Use the `libtcl.cfg` files to specify an LDAP server name, port number, DIT base, user name, and password to connect to an LDAP server.

In its simplest form on 32-bit LDAP drivers, the `libtcl.cfg` file is in this format:

```
[DIRECTORY]
ldap=libsybdldap.dll
```

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

Using a Directory Service

For more information, see the configuration guide for your platform.

1. Configure the `libtcl.cfg` file to use directory services—use any standard ASCII text editor to:
 - Remove the semicolon (;) comment markers from the beginning of the LDAP URL lines in the `libtcl.cfg` file under the `[DIRECTORY]` entry.
 - Add the LDAP URL under the `[DIRECTORY]` entry. See the *Configuration Guide* for supported LDAP URL values.

Warning! Enter the LDAP URL on a single line.

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

For example:

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

Note: On Windows x64, the `.dll` file is called `libsybdldap64.dll`.

2. Verify that the appropriate environment variable points to the required third-party libraries. The Netscape LDAP SDK libraries are located in `%SYBASE%\%SYBASE_OCS%\dll`

The Windows PATH environment variable must include this directory.

3. Use **dsedit** to add a server to the directory service:
 - a) From Windows, select **Start > Programs > Sybase > Connectivity > Open Client Directory Service Editor**.
 - b) Select **LDAP** from the list of servers, and click **OK**.
 - c) Click **Add New Server Entry**.
 - d) Enter:
 - The server name – this is required.
 - Security mechanism – optional. A list of security mechanism OIDs are located in %SYBASE%\ini\objectid.dat.
 - e) Click **Add New Network Transport**.
 1. Select the transport type
 2. Enter the host name
 3. Enter the port number
 - f) Click **OK** twice to exit the **dsedit** utility.

Upgrade

You can upgrade to Adaptive Server 15.5 from Adaptive Server versions 15.0 through 15.0.3 (except 15.0.1 Cluster Edition and 15.0.3 Cluster Edition).

If you are installing Adaptive Server 15.5 on top of an earlier 15.x version see Installing version 15.5 over an existing 15.x Adaptive Server.

If you need information on recovering from a failed upgrade see Recovering from a failed upgrade.

Note: Adaptive Server 15.5 is certified for the operating system versions described in the System Requirements section of this document. Adaptive Server 12.5.x is certified for the same operating system versions. Hence, an installation upgrade from 12.5.x release may be made to a 15.x version of Adaptive Server.

Note: Adaptive Server version 15.5 does not support upgrade from Adaptive Server version 12.5.3a.

Adaptive Server version 15.5 introduces some new system catalogs and some changes to existing catalogs.

The catalog upgrade in 15.5 may affect your existing applications. If you are upgrading, see the *Adaptive Server Enterprise New Feature Guide for 15.5* for a complete list of catalogs that are affected.

For a server installation 12.0.x and older, Sybase recommends that you upgrade to version 12.5.4, then upgrade to version 15.5.

You can upgrade Adaptive Server from a 32-bit version to a 64-bit version, but you cannot move from a 64-bit version to a 32-bit version. Likewise, you can upgrade only from an earlier version of Adaptive Server to a more recent version.

Only upgrades to and from the same page size are supported. Changing the server schema from one page size to another is a database migration. See **sybmigrate** in the *Utility Guide* for information about migration.

Preparing to Upgrade

You must have system administrator privileges to perform an upgrade.

Note: If your server has replicated databases, consult *Upgrading Servers with Replicated Databases*, before starting the preupgrade tasks.

Also, if you are upgrading from Adaptive Server 12.5.4 or earlier, execute the upgrade sequence from the same node on which the database resides.

If your server is already at a 15.x level, do not upgrade using **sqlupgrade** or **sqlupgraderes**. The message “Adaptive Server is already at version 15.0” appears if you are already at a 15.0 level. All required changes happen automatically while the first time you start the new server. The **preupgrade** and **upgrade** utilities are internally called by **sqlupgrade** or **sqlupgraderes**.

Each new server version contains features that introduce new parameters, commands, reserved words, and so on. **preupgrade** prepares the old server for upgrade by checking to make sure all necessary directories and settings are correct for the upgrade. When running **preupgrade**, manually stop and start the server. It is not necessary to start the server before running **sqlupgrade**. **sqlupgrade** starts the server when needed.

1. Install Adaptive Server 15.5 into its own installation directory.
2. If you are upgrading from a 12.5.3 installation with cache bindings on **sybssystemdb**, before running **preupgrade**, drop the cache bindings for **sybssystemdb** that are bound to a user-defined cache.
3. If your server is a pre-15.x version then, from the 15.5 server installation, perform the **preupgrade** checks on the old server using the **preupgrade** utility, which is located at `$(SYBASE)/ASE-15_0/upgrade`. If necessary, fix any problems that **preupgrade** process reports. If your server is already at Adaptive Server 15.x or later most of this process is automatic when you install the new server.
4. When **preupgrade** runs without reporting any errors, you can then:
 - a) Shut down the pre-15.x server.
 - b) Copy the server entry from the old `interfaces` file to the new `interfaces` file.
 - c) copy the `<server_name>.cfg` file from old directory to the new `$(SYBASE)` directory.
 - d) Start the 15.5 server with the old master device from the old server.
 - e) Run the upgrade process. If problems occur, fix them and then re-run the upgrade process.
 - f) Reinstall the Sybase-supplied stored procedures to prevent any errors caused from changes to the system catalogs.

Note: Adaptive Server issues advisory warnings when configuration parameters are not set to the default. No action is required.

5. Following an upgrade to other platforms, the server internally rebuilds procedures from the text in `syscomments` the first time the server is executed. The procedure may not execute properly if it contains code that drops an existing object and recreates it. This is a problem when the object already exists at the time the procedure is first executed. To avoid problems, manually drop the object prior to executing the procedure for the first time.

As part of the **preupgrade** tasks, **preupgrade** scans all databases and catalogs and determines how much free space is required to upgrade successfully. **preupgrade** searches for the largest catalog, then calculates the required free space by doubling the size of the largest catalog, and adding 10 percent for logging the upgrade changes for each catalog.

Preupgrade Tasks

Learn necessary preupgrade tasks here.

You need the master and system procedures files to upgrade the server. By default, the *master* (*master.dat*) and system procedures device (*sybsystemprocs.dat*) files are installed in the `$SYBASE/data` directory.

Note: If you are upgrading Adaptive Server, the previously installed version of the server must be running. If you are upgrading Backup Server, Historical Server, Monitor Server, or XP Server, those servers must not be running.

1. Ensure that the directory to which your TEMP environment variable points exists. The installation program uses the directory to write files temporarily during the installation, which it locates through the TEMP environment variable.
2. Verify the name and location of the `runserver` file.
3. Verify that the `runserver` file is renamed `RUN_servername`, where *servername* is the name of the old server.

The *servername* must appear exactly as it appears in the `interfaces` file. The `RUN_servername` file for a server SYBASE is called `RUN_SYBASE`. If the `RUN_servername` file for your current server is named differently, change the name during the upgrade process.

4. To upgrade stored procedures, the stored procedure text must be available in *syscomments*. Do one of the following:
 - Reinstall the procedures with text.
 - Remove them and reinstall them after the upgrade.
5. Set the password policy option '**allow password downgrade**' to 1. Then passwords are encrypted with old and new algorithms aiding in transition should you ever downgrade to an earlier version.
6. Reserved words are resolved using quoted identifiers.
7. Verify that users are logged off.
8. Check database integrity.
9. Back up databases.
10. Dump the transaction log.
11. Make sure that master is the default database for the "sa" user.
12. Prepare the database and devices for upgrade using the **preupgrade** utility.
 - a) Create a *sybssystemdb* database if it does not already exist.
 - b) Disable auditing using the command:


```
sp_configure 'auditing', 0
```
 - c) Save the current audit settings for the pre-15.5 Adaptive Server using the command:

```
sp_displayaudit
```

13. Disable disk mirroring.

14. Verify that your SYBASE environment variable points to the location of the new server software files you just installed.

Set the OLDSYBASE, OLDSYBASE_ASE, and OLDSYBASE_OCS environment variables to the location of the server you are upgrading, to avoid having to type this path when you run **sqlupgrade**.

15. Disable Job Scheduler.

16. Ensure that the procedure cache size is at a minimum of 150 percent of the default procedure cache size, or between 53248 and 2147483647 2K pages.

17. Copy the following files from earlier versions of the server to their corresponding Adaptive Server 15.x installation location.

- \$SYBASE/interfaces
- \$SYBASE/\$SYBASE_ASE/<servername>.cfg
- \$SYBASE/\$SYBASE_OCS/config/libtcl.cfg
- \$SYBASE/SYSAM-2_0/licenses/license.lic

18. If the Java in the database feature is enabled, create the **sybpcidb** database. The **sybpcidb** database stores configuration information for the Pluggable Component Interface (PCI) and all Pluggable Component Adapter (PCA) components. This database is used by **installpcidb** script. For example:

```
1> disk init
2> name = "sybpcidb_dev",
3> physname = "${SYBASE}/data/sybpcidb_dev.dat",
4> size = '24MB'
5> go
1> create database sybpcidb on sybpcidb_dev = 24
2> go
```

19. Set the environment variables by sourcing the SYBASE.[csh, sh, env]. file.

Running the environment variable script multiple times can cause problems. Only run it once.

Reserved Words

Reserved words are pieces of SQL syntax that have special meaning when used as part of a command.

Transact-SQL does not allow words that are part of command syntax to be used as identifiers, unless they are enclosed in quotation marks. If you are upgrading Adaptive Server, and the identifiers in your user databases match new reserved words, errors result when you run queries, stored procedures, or applications that use these identifiers.

Note: Before performing your upgrade, change the name of any user database that is a reserved word using the **sp_renamedb** stored procedure.

Conflicts between object names do not prevent the upgrade process from completing. However, applications that refer to conflicting object names may not work after the upgrade. Rename all objects that use reserved words.

As part of the pre-upgrade process, **sqlupgrade**, **sqlupgraderes**, or **preupgrade** can perform the reserved word check for you. See the *Reference Manual* for a complete list of reserved words.

Note: If you change an object name, change applications and stored procedures that refer to that object.

Running a Reserved Word Check

If you plan to use **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.

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.

Both **sqlupgrade** or **sqlupgraderes**, automatically install new reserved words and **sp_checkreswords**. **sp_checkreswords** is a stored procedure used to detect and display identifiers in your existing databases that conflict with reserved words in the new database. You can then run the new version of **sp_checkreswords** at any time while performing pre-upgrade tasks.

Note: You must use the **sp_checkreswords** stored procedure from the new version of Adaptive Server to verify that the old installation of Adaptive Server does not use any reserved words that have been introduced with the new server.

1. 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/sqlupgrade MMDD.nnn`. Review this file to determine which identifiers must be changed.
2. The **sp_checkreswords** system procedure detects and displays any identifiers in your existing databases that conflict with reserved words. Server Config installs **sp_checkreswords** and checks the reserved words during the upgrade. **Server Config** does not display the names or locations of identifiers that conflict with reserved words, only the number of conflicts.

Addressing Reserved Words Conflicts

use these techniques to address reserved word conflicts

- If any database names are reserved words, you must use **sp_renamedb** to change the database names before you can upgrade. Use **sp_dboption** to set the database to single-

user mode, and then run **sp_renamedb**, specifying the new name. See the *Reference Manual* for more information on these procedures.

- If other identifiers are reserved words, you can use:
 - **sp_rename** to change the name of the object, before or after the upgrade.
 - Quoted identifiers.
 - Brackets around the identifier. For example:

```
create table [table] ( [int] int, [another int] int )
```
- Run **sp_checkreswords** in *master* and in each user database to display the names and locations of conflicting identifiers.

For more information about **sp_rename** and **sp_checkreswords** and methods for avoiding reserved word conflicts, see the *Reference Manual*.

Using Quoted Identifiers

The **set quoted_identifier** option tells Adaptive Server to treat any character string enclosed in double quotation marks as an identifier.

- 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.
- 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.

Preparing the Database and Devices for the Upgrade

If you ran the pre-upgrade option in **preupgrade**, the utility scanned the system catalogs and calculated how much additional space needed to perform the upgrade.

Note: Sybase recommends that you use **preupgrade** to determine how much free space is required for the upgrade.

1. If you ran the pre-upgrade option in **preupgrade**, the utility scanned the system catalogs and calculated how much additional space needed to perform the upgrade.
2. If you did not run the pre-upgrade check, you must manually calculate how much free space you must add to your system catalogs and databases.

As a general rule, you must double the size of the largest catalog that you are going to upgrade, and add approximately 10 percent more to calculate the free space required for the upgrade to succeed. For example, if you have a large number of stored procedures or compiled object in any database, the *syscomments* and *sysprocedures* catalogs require additional space.

3. If you do not have a *sybsystemdb*, and are not using two-phase commit or DTM applications, create a *sybsystemdb* with a minimum of 4MB. If you are using two-phase commit or DTM, you can either:
 - Create a minimum-sized *sybsystemdb* for upgrade and expand it later, or
 - Create a *sybsystemdb* of 5 to 20MB, depending on your expected usage. Approximately 25 percent of the database should be data storage, and 75 percent should be log storage.

Adaptive Server uses this database for tracking transactions and during recovery. In addition, it is used for applications using two-phase commit and Distributed Transaction Management (DTM).

4. If you do not have a *sybsystemdb* database:

```
1> use sybsystemdb
2> go
1> sp_create_syscoordinations
2> go
```

5. If you have a *sybsystemdb* and the database is at least 4MB, you do not need to make any changes. If the database is smaller than 4MB, increase the size with **alter database**.
6. Verify that the *sybsystemprocs* database is large enough. For an upgrade, the default size for *sybsystemprocs* is 124MB, or enough free space to accommodate the existing *sybsystemprocs* database, and the largest catalog that is to be upgraded, plus an additional 10 percent for logging upgrade changes. You may need more space if you are adding user-defined stored procedures. 124MB accommodates additional internal data structures, but does not account for the possibility of a larger number of user-defined system procedures.
7. If you are using the Java in the database feature you must enable PCI and configure **sybpcidb** database. For **sqlupgrade[res]**, the **enable java** setting in the old server is detected. If it is set at 1, then **preupgrade** asks you to create a **sybpcidb** device and database before you run **upgrade**.
8. Increasing the default database size. See the Increasing Default Database Sizes topic for instructions on how to do this.

Increasing Default Database Sizes

You can use **sp_helpdb** or Sybase Central to determine the current size of the *master*, *model*, *sybsystemprocs*, and *tempdb* databases.

- 1.

In **isql**, use **alter database** to increase the size of the *master* database. For example:

```
1> use master
2> go
1> alter database master on master=x
2> go
```

This example increases the size of the *master* database; *x* is the number of megabytes of space added to the existing database size.

2.

Repeat this step to increase the size of each of the *tempdb* and the *model* databases. For example:

```
1> alter database tempdb on master=x
2> go
1> alter database model on master=x
2> go
```

Note: The *model* database cannot be larger than the *tempdb*. If your changes would cause the *model* database to be larger than the *tempdb*, then you must first increase the size of *tempdb*, before increasing the size of the *model* database.

3. Verify the size of each database. Enter the following, where **database_name** is the name of the system database you are checking:

```
sp_helpdb database_name
```

4. Use **sp_configure** to update the values as indicated by the **preupgrade** utility. In this example "x" is the number suggested by the **preupgrade** utility.

```
1> sp_configure "number of locks", x
2> go
```

Upgrading to Adaptive Server 15.5

To assist with the upgrade process, Sybase recommends you set the `OLDSYBASE`, `OLDSYBASE_ASE`, and `OLDSYBASE_OCS` environment variables to point to the old installation of Adaptive Server. If you set these environment variable to the old installation of the Adaptive Server, the upgrade utility automatically populates input fields with the correct information as you upgrade.

1. Set the `OLDSYBASE_ASE` variable to the `SYBASE_ASE` appropriate to your older server. If you are upgrading from 12.5 for example, it should be `ASE-12_5`.
2. If upgrading from Adaptive Server 12.5:

```
setenv OLDSYBASE <old_$SYBASE_directory>
setenv OLDSYBASE_ASE ASE-12_5
```

This is required if you are using the **sqlupgraderes** utility to perform your upgrade.

3. 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`.
- **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 the Using Sqlupgraderes topic.

Using Sqlupgrade

sqlupgrade verifies that the server is eligible for the upgrade. Perform all pre-upgrade tasks before running **sqlupgrade**.

1. Make sure your old server is running.
2. Verify that you have performed the tasks mentioned on the window, before beginning the upgrade.
3. Source `SYBASE . csh` before running **sqlupgrade** if you have not already done that.
- 4.

Execute **sqlupgrade**, by entering at the UNIX prompt:

```
$SYBASE/$SYBASE_ASE/bin/sqlupgrade
```

A **sqlupgrade** window now displays.

5. The first screen warns you to back up the databases, devices, and so on. Click Next to proceed to the next screen.
6. Click **OK**.
7. The Next screen prompts you to enter the old Sybase directory, and the old Adaptive Server directory.
8. Click **OK**. The Adaptive Server Selection window displays.
9. 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.

10. Click **OK**. The SA Password window is displayed.
11. Enter the password for the System Administrator (sa) login.
12. Click **OK**. The Specifying Upgrade Options window is displayed.
13. Specify an upgrade option.

The upgrade eligibility tests verify that your current server meets the requirements for upgrading to the new version. (The online help provides a partial list of what the upgrade eligibility tests check.) If your current server fails the upgrade eligibility test, you must fix the problem.

14. Click **OK** to upgrade your server.

sqlupgrade displays the Status Output window. View the Status Output window 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`. *NNN* is the month, *DD* is the date, and *NNN* is a three-digit number identifying the srbuild session.

After all pre-upgrade checks are successful, the old server is shut down and the new Adaptive Server *dataserver* binary is started on the old *master* device. **sqlupgrade** internally runs the upgrade binary to perform the upgrade.

If the upgrade is successful, a “Done” message is displayed in the Status Output window.

15. The upgrade process:

- Creates the `RUN_servername` file
- Runs the `installmaster` script

16. Check that the upgrade has been successful by logging in to the server and checking `select @@version` to confirm that it is at 15.5. Also, **sp_configure “upgrade version”** should return 15000.

17. For information about compiled objects, see Troubleshooting.

When the upgrade completes successfully, you can:

- Click OK to specify another server to upgrade, or
- Exit **sqlupgrade**, and go to Post-upgrade tasks.

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.

1. Edit the sample resource file included in your Adaptive Server distribution in `$SYBASE/$SYBASE_ASE/init/sample_resource_files` directory. Use an ASCII text editor to edit the resource file as described in Alternative Installation Methods.
2. Run the **sqlupgraderes** utility. This utility creates a new file containing the values you specified for the upgrade session, and writes it to the `sqlupgrademmdd.NNN-server_name.rs` file in `$SYBASE/$SYBASE_ASE/init/logs/`.

Where *server_name* is the name of the upgraded server, *MM* is the month, *DD* is the date, and *NNN* is a three-digit number identifying the srbuild session.

For details on the attributes used in a resource file, see Resource file attributes for upgrading Adaptive Server, below.

Resource File Attributes for Upgrading Adaptive Server

Look here to learn about 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 `$SYBASE/$SYBASE_ASE/init/sample_resource_files` file included in your Adaptive Server distribution shows examples of these attributes.

All attributes are required, and all values are case sensitive.

If the resource file you are modifying was created by **sqlupgrade**, the prefix of the attribute name may vary. **sqlupgraderes** ignores this prefix when processing the resource file.

Attribute	Default value [other options]
sybinit.release_directory	\$SYBASE [<i>path = _name_of_old_release</i>]
sybinit.product	sqlsrv
sqlsrv.server_name	<i>server_name</i>
sqlsrv.new_config	No
sqlsrv.sa_login	<i>current_login</i>
sqlsrv.sa_password	<i>current_password</i>
sqlsrv.do_upgrade	Yes
sqlsrv.do_reserved_word_check	Yes

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.

1. Source the version 15.5 SYBASE .csh file before running **sqlupgraderes** and set OLDSYBASE and OLDSYBASE_ASE.
2. To execute **sqlupgraderes**, at the UNIX prompt, enter the following, where **resource_file** specifies the resource file containing the attributes that describe the server to upgrade:

```
$SYBASE/$SYBASE_ASE/bin/sqlupgraderes -r resource_file
```

Enabling PCI in Adaptive Server

To enable Java in the database feature you must enable PCI and configure **sybpcidb** database.

For **sqlupgrade[res]**, the **enable java** setting in the old server is detected. If it is set at 1, then **preupgrade** asks you to create a **sybpcidb** device and database before you run **upgrade**.

Postupgrade Tasks

After you have upgraded to the new version of Adaptive Server, perform these tasks to make sure your new Adaptive Server is up and running.

Note: After the upgrade, you do not need to run **update statistics** on any tables. The upgrade process does not cause any changes in existing statistics. However, if you are upgrading from 15.x version, restart the server to make the statistics available.

1. If you upgraded from Adaptive Server version 12.5.2 or lower, after the upgrade you must run **dbcc checkcatalog** with the **fix** option to ensure there are no issues with *OAM pages*.

```
dbcc checkcatalog (database_name, fix)
```

2. The new system stored procedures must be installed before any application activity starts. The scripts can be installed using the following **isql** commands:

```
isql -Usa -P<sa_password> -S<server_name>  
-i$SYBASE/$SYBASE_ASE/scripts/<script_name> -o<output_file>
```

The output of the script is saved in **output_file**. If you are using **dbcc upgrade_object**, Sybase recommends that you read the **Upgrading compiled objects with dbcc upgrade_object** topic in the **Troubleshooting the Server** section of this book.

3. If you have modified the system stored procedures without altering their original names, executing some of these scripts overwrites your modifications. You must reload the modified stored procedures after running these scripts:
 - **installcommit** – If you use two-phase commit or distributed transactions, you must rerun the **installcommit** SQL script to restore the following stored procedures:
 - **sp_start_xact**
 - **sp_abort_xact**
 - **sp_remove_xact**
 - **sp_stat_xact**
 - **sp_scan_xact**
 - **sp_probe_xact**
 - **installsecurity** – Run this script if the Adaptive Server Auditing functionality was enabled in your earlier installation.
 - **installhasvss** – Run this script if the HA functionality is enabled and the HA feature is currently in use in the upgraded installation of Adaptive Server.
 - **installmsgsvss** – Run this script if the **RTDS ASE MESSAGING** functionality is enabled and the **RTDS ASE MESSAGING** feature is being used in this installation.
 - **installpcidb** – Run this script if Java in the database feature was enabled in your earlier installation.

- **installjsdb** – Run this script if Job Scheduler functionality was enabled in your earlier installation.

Running the Instmsgs.ebf Script

Follow these instructions if you are upgrading from an Adaptive Server version 15.0.x to Adaptive Version 15.5.

Adaptive Server restores error messages back to the 15.0.x release from which you are upgrading. Before installing the new messages for 15.5 you must run **uninstmsgs.ebf** to safeguard the modified messages in your master device.

Note: If you are upgrading from a pre 15.x release, then ignore this step to produce the `restore_msgs.ebf` file and proceed to run the **instmsgs.ebf** file.

1. To run **uninstmsgs.ebf** use the following command:

```
isql -Usa -P<sa-password> -w1000 -iuninstmsgs.ebf -
orestoremsgs.ebf
```

2. Run the **instmsgs.ebf** script after you have completed running the **uninstmsgs.ebf** script. Use the following syntax to run **instmsgs.ebf**:

```
isql -Usa -P<sa-password> -iinstmsgs.ebf
```

3. If you are a localized customer, run the **instmsgs.ebf** command, and then install any localized language using **langinstall**, **sqlloc**, or **syconfig**.

- a) If you need to undo the **instmsgs.ebf** changes, run the following script once you have downgraded to the version from which you upgraded.

```
isql -S -Usa -P <restore_msgs.ebf>
```

If you run **instmsgs.ebf** after installing the 15.5 localized message, this script may delete some new messages.

Restoring Functionality in Adaptive Server

Learn how to restore functionality in the server after an upgrade.

1. Reset the configuration parameters.

If you changed any configuration parameters before upgrading, use **sp_configure** to set them back to their earlier values.

2. Use **sp_dboption** to reset any database options you disabled before upgrading.
3. After you complete the upgrade, but before you use the upgraded server, verify that all scripts developed at your site point to Adaptive Server 15.5.
4. Verify procedure cache allocation after upgrading.

This section is for information only. After the upgrade Adaptive Server ensures that **procedure cache size** is the same as before upgrade, unless the original size was less than the default value.

If you are upgrading from a version of Adaptive Server that is earlier than version 12.5.x, **procedure cache percent** is changed to **procedure cache size**. Adaptive Server now allocates memory dynamically and determines the size of the procedure and data caches in terms of megabytes. During the upgrade process, the server converts the **procedure cache percent** to an absolute value and sets **procedure cache size** based on this value.

For example, if **procedure cache percent** is configured as 20 percent of 200MB of memory, then the absolute value of **procedure cache percent** is 40MB. During the upgrade process, the server converts **procedure cache percent** at 20 percent to **procedure cache size** at 40MB. If the value for **procedure cache size** is less than the default value, the server sets **procedure cache size** to the default value. For example, if **procedure cache percent** is set to 10 percent in the example above, and the absolute value of 10 percent is less than the default value of **procedure cache size**, then **procedure cache size** is set to the default value.

With the introduction of dynamically reconfigured memory configuration parameters in Adaptive Server 12.5.x, an increase in the server's memory use does not decrease the size of the procedure cache or the data cache. That is, if your default data cache is set to 40MB, and you increase your procedure cache, the default data cache is still 40MB.

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

5. Check the procedure cache requirements.

Stored procedures, triggers, and other compiled objects require much more memory to run Adaptive Server 15.0.x than older versions.

You may increase the **procedure cache size** during runtime, using **sp_configure**. You can use the **verify** option in **sp_configure** to verify any changes you make to the configuration file without having to restart Adaptive Server. The syntax is:

```
sp_configure "configuration file", 0, "verify",  
"full_path_to_file"
```

For example:

```
sp_configure "configuration file", 0, "verify",  
"/work2/Sybase/ASE125.cfg"
```

For more information about **sp_configure** and **sp_sysmon**, see the *Reference Manual* and the *Performance and Tuning Guide*. For more information about configuring memory, see the *System Administration Guide*.

6. Verify data cache allocation after upgrading.

The server ensures that all the data cache sizes after an upgrade, are the same as before the upgrade. Adaptive Server 12.5.x and later treats the default data cache size as an absolute value and sets it in the `config` file. The string `DEFAULT` in the `config` file has a different meaning for default data cache size in Adaptive Server 12.5.x and later.

In pre-12.5 Adaptive Server, it meant all the memory left after allocating memory for other configurations, procedures, and user-defined named caches. In Adaptive Server 12.5.x and later, it means a default value of 8MB.

During the upgrade process, the server ensures that the default data cache size remains the same. Therefore, during the preupgrade process, the size of the default data cache is obtained and written to the configuration file as an absolute value, not as a “DEFAULT”. This enables the server to have the same default data cache size as before the upgrade. If this size is less than the default size of default data cache of 8MB, then the server allocates a default data cache of size 8MB.

7. If you unmirrored devices, remirror them, using the **disk remirror** command.
8. For information about upgrading compiled objects, see [Upgrading compiled objects with dbcc upgrade_object](#).
9. If you used two-phase commit in 11.9.x, run the script to install the two-phase commit tables:

```
isql -Usa -Psa_password -Sserver_name
-i$SYBASE/$SYBASE_ASE/scripts/installcommit
```

You do not need to run this script if you do not use two-phase commit.

Reenabling Replication Server After the Dataserver Upgrade

If you disabled replication before the upgrade, you must reenabling replication.

1. Remove older formatted log records from your database.
2. Use the **dump tran** command to dump the database and transaction logs to remove the older format log records from your database. This prevents Replication Agent™ and other log readers from accessing the pre-upgrade portion of the transaction logs.

```
1> use master
2> go
1> dump database sales to dumpdev
2> go
1> dump transaction sales with truncate_only
2> go
```

3. Reenable replication

Restore Replication on Destination Databases

Restore replication if you have upgraded a destination only dataserver in a replication system. If the target database is also a source for replication, follow the steps in [Restoring Replication on Primary Databases](#).

For each primary database and RSSD:

1. Start Adaptive Server if it is not already running.

2. Log in to the server.
3. If you zeroed out the locator point before the upgrade you don't need to execute this step.

Otherwise, clear the locator for the database by executing the following command in the RSSD for this database:

```
1> use RSSD
2> go
1> rs_zeroltm dataserver, database
2> go
```

Warning! You must stop Replication Server before executing `rs_zeroltm`

4. Restart the Replication Servers.
5. 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 15.0.x, 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 *Installing Sybase PC-Client Products* or the *Open Client Configuration Guide*.

Restoring Replication on Primary Databases

Follow the steps in this section if you have upgraded a source database, or primary database, in a Replication Server system, or the destination database is also a source for other Replication Servers.

1. If you zeroed out the locator point before the upgrade or you have already resumed replication on the earlier section, you don't need to execute this step.

Otherwise, clear the replication locator by executing the following command in the RSSD database: Log in to the RSSD for each replicated primary, and for each replicated RSSD, and issue:

```
1> use RSSD_name
2> go

1> rs_zeroltm dataserver, database
2> go
```

Warning! You must stop Replication Server before executing `rs_zeroltm`

2. Log in to each replicated primary, and replicated RSSD and issue:

```
1> use database
2> go
```

```
1> dbcc settrunc ('lrm', 'valid')
2> go
```

3. Restart the Replication Server.

4. If the database is also used as an RSSD, resume the Replication Server connection to the RSSD by issuing the following command to the Replication Server specifying the same string that was specified during the **'hibernate_on'** command:

```
1> sysadmin hibernate_off, 'Replication Server'
2> go
```

5. Log in to the Replication Server, and resume the Log Transfer connection for each replicated primary and each replicated RSSD:

```
1> resume log transfer from server.database
2> go
```

If this is a replicated RSSD, you must log in to the replicate Replication Server.

6. If you are using Rep Agent, log in to the Adaptive Server and restart the Rep Agent:

```
1> use database
2> go
1> sp_start_rep_agent database
2> go
```

7. If you are using LTM, restart the LTM.

Reenabling Auditing

If the server you are upgrading from was configured for auditing, reenabling auditing in the upgraded server.

1. using this command to reenabling auditing:

```
sp_configure 'auditing', 1
```

2. Installing system stored procedures of old version drops audit information about the system stored procedures. Reenable auditing for all system stored procedures for which auditing was enabled before the upgrade.

a) Use the output of **sp_displayaudit** recorded during pre-upgrade to help determine the system stored procedures for which auditing was enabled.

b) Reenter the audit options using **sp_audit**. For example, if you had enabled stored procedure auditing for **sp_addlogin** stored procedure in your server before upgrade, run **sp_audit** to reenabling auditing on **sp_addlogin**:

```
sp_audit "exec_procedure", "all", "sp_addlogin", "on"
```

Special Considerations for Threshold Procedures for Audit Segments

Learn about special considerations for threshold procedures for audit segments.

If your installation used a threshold procedure to archive the *sysaudits* table using T-SQL similar to the following example:

Upgrade

```
INSERT MyPre15SysAuditHistoryTable SELECT * FROM  
sysaudits_0n
```

Where *n* corresponds to the *sysaudits* table number 1 – 8 and *MyPre15SysAuditHistoryTable* is a table defined prior to Adaptive Server version 15.5, then you must alter *MyPre15SysAuditHistoryTable* to add a *nodeid* column using the following command:

```
alter table MyPre15SysAuditHistoryTable  
add nodeid tinyint NULL
```

See the *Reference Manual* for system tables *sysaudits01* through *sysaudits08* for further details about the *sysaudits* tables.

Upgrading Backup and Monitor Servers

Before you can upgrade Backup or Monitor Server, you must upgrade Adaptive Server. The procedures for upgrading Backup Server and Monitor Server are similar. XP Server does not have a formal upgrade process.

Prerequisites

- If you are upgrading during the initial server installation, select Upgrade Existing Servers from the drop-down menu when the installer prompts Build or Upgrade Adaptive Server, Backup, and Monitor Servers.

This launches the **sqlupgrade** utility. Click **OK**.

- If you are upgrading after the initial installation, start the **sqlupgrade** utility from the command line. Enter:

```
$(SYBASE)/$(SYBASE_ASE)/bin/sqlupgrade
```

Task

1. A warning message appears. Click **OK** to continue.
2. The next screen prompts you to enter the old Sybase directory and the old Adaptive Server directory.
3. From the Server selection window, select the server to upgrade. Click **OK**.
4. When you are prompted for the password, enter the password for the System Administration, and click **OK**.
5. The Status output window display the upgrade progress. When the upgrade is complete it displays, “Done.” Click **OK**.
6. You return to the **sqlupgrade** window. You can either upgrade another server, or click **Exit** to leave the utility.

Repeat this procedure to upgrade Monitor Server.

Upgrading Java in the Database

You must run **installpcidb** script if the Java in the database functionality is currently enabled. This script creates tables and stored procedures in the **sybpcidb** database.

1. Using **isql**, run the **installpcidb** script that is included with the release. Save the output of this step to an operating system file.

```
isql -Usa -P<sa_password> -S<server_name>
-i$SYBASE/$SYBASE_ASE/scripts/installpcidb -o<output_file>
```

2. Enable the following two configuration parameters in order to enable the Java in the database functionality.

```
1> sp_configure 'enable pci', 1
2> go
1> sp_configure 'enable java', 1
2> go
```

You may need to increase the **'max memory'** in order to enable these parameters. After they are enabled it is necessary to restart the server for changes to take affect. You can configure the maximum size of the PCI Bridge memory pool through the **'pci memory size'** configuration parameter. For more information see *Java in Adaptive Server Enterprise for 15.5*.

Enabling the Java in The Database Feature in a High Availability System

The Java in the database feature may be used on a system that also uses high availability.

Drop high availability companionship prior to the installation of the **sybpcidb** and re-established the companionship afterwards.

Java in the database feature must be either enabled or disabled on both nodes of the high availability system.

Upgrading Job Scheduler

After upgrading to the new Adaptive Server, perform the following steps to upgrade Job Scheduler.

Note: You must have `$SYBASE/$SYBASE_OCS/bin` directory in `$PATH` so the **isql** executable is accessible.

1. Copy the directory services entry for JSAGENT (or jsagent) from the old server to the new 15.5 server.
2. Make sure the 15.5 server is running.

3. Stop the Job Scheduler. From **isql**, enter:

```
1> sybmgmtdb..sp_sjobcontrol @name=NULL, @option="stop_js"  
2> go
```

4. Ensure that at least 9000 locks are configured. If the number of locks on the server is less than 9000, use **sp_configure** to increase the number of locks:

```
1> sp_configure "number of locks", 9000  
2> go
```

5. Upgrade the internal Job Scheduler SQL using **isql**, enter:

```
1> use sybmgmtdb  
2> go  
1> dbcc upgrade_object  
2> go
```

6. Restart Adaptive Server.

7. If the *sybmgmtdev* device is smaller than 100MB, increase the size to at least 100MB. From **isql** enter the following, where *nn.nM* is the number of megabytes needed to get to 100 MB:

```
1> use master  
2> go  
1> disk resize name = 'sybmgmtdev' , size = 'nn.nM'  
2> go
```

8. (Optional) Add more log space. Some 64-bit platforms require additional space for the *sybmgmtdb* log. From **isql**, enter:

```
1> use master  
2> go  
1> alter database sybmgmtdb LOG on sybmgmtdev=20  
2> go
```

9. To upgrade *sybmgmtdb* using **isql**, run the `installjsdb` script that is included and save the output to a file:

```
isql -Usa -Psa_password -Sservername -n -i$SYBASE/$SYBASE_ASE/  
scripts/installjsdb  
-ooutput_file
```

10. Enable Job Scheduler to start when Adaptive Server starts. From **isql**, enter:

```
sp_configure "enable job scheduler", 1
```

11. To start Job Scheduler, from **isql**, enter:

```
sybmgmtdb..sp_sjobcontrol @name=NULL, @option="start_js"  
go
```

Upgrading Job Scheduler Templates

After upgrading to the new server, upgrade the templates and jobs created by Job Scheduler templates.

Note: Several changes impact the Job Scheduler templates. These changes make some templates incompatible with earlier versions of the server. The current templates are version 3.0 in the XML files.

1. Disable Job Scheduler before upgrading the templates.
2. Update any environment variables, scripts, or applications that refer to the Job Scheduler directory path. The Job Scheduler directory has been renamed and moved beneath the ASE-15_0 directory. The new location is `$$SYBASE%\$SYBASE_ASE/jobscheduler`.

Directories below `jobscheduler` remain the same.

3. Copy files from the `jobscheduler` directory to the new ASE-15_0 directory. If you are installing the new server directories over the old server directories, the installer moves your files to the new `jobscheduler` directory automatically.
4. If you modified Sybase-supplied templates, stored procedures, or XML documents, do not overlay the new templates with your modified versions. If you do, you lose the enhancements provided by the revised templates. Carefully merge your template changes into the Sybase template files, or better yet, rename your modified templates.

Note: If you modify Sybase-supplied templates, save the changes under new names and in new files.

5. You may need to make minor modifications to jobs you created from 2.0 or earlier templates. In some templates, parameters changed from `varchar(5)` to `int`. The following table lists the templates that changed in version 2.1 and the changes you must make to jobs created from them.

Table 11. Changed Job Scheduler Templates

Template	Modified files	Change description	Modify jobs created from this template
dump database	SybBackupDbTo-DiskTemplate.xml jst_dump_databases	Changed the datatype for the @use_srvr_name parameter from <code>varchar(5)</code> to <code>int</code>	Required, modify job SQL to change @use_srvr_name to <code>int</code> datatype.
dump database log	SybBackupLogTo-DiskTemplate.xml jst_dump_log	Changed the datatype for the @truncate_flag and @use_srvr_name parameters from <code>varchar(5)</code> to <code>int</code>	Required, modify job SQL to change @truncate_flag and @use_srvr_name to <code>int</code> datatype.
update statistics	SybUpdateStatsTemplate.xml jst_update_statistics	Changed the datatype for the @index_flag parameter from <code>varchar(5)</code> to <code>int</code>	Required, modify job SQL to change @index_flag to <code>int</code> datatype.

Template	Modified files	Change description	Modify jobs created from this template
rebuild indexes	SybRebuildIndex-Template.xml jst_reorg_rebuild_indexes	Changed the datatype for the @dump_flag parameter from <i>varchar(5)</i> to <i>int</i>	Required, modify job SQL to change @dump_flag to <i>int</i> datatype.
rebuild table	SybRebuildTable-Template.xml jst_reorg_rebuild_tables	Changed the datatype for the @dump_flag parameter from <i>varchar(5)</i> to <i>int</i>	Required, modify job SQL to change @dump_flag to <i>int</i> datatype.
reclaim indexes	SybReclaimIndex-Template.xml jst_reclaim_index_spac	Required, modify job SQL to change @dump_flag to <i>int</i> datatype.	Required, modify job SQL to change @resume_flag to <i>int</i> datatype.
reclaim tables	SybReclaimTable-Template.xml jst_reclaim_table_space	Changed the datatype for the @resume_flag parameter from <i>varchar(5)</i> to <i>int</i>	Required, modify job SQL to change @resume_flag to <i>int</i> datatype.

6. Some Job Scheduler templates were modified to support the new server functionality. These changes are related to new parameters for specifying a partition name or a datachange value for the new server commands that added these options. If you have jobs created from any of the enhanced templates, modify the SQL of the jobs for the 15.5 servers.

Note: If you have a job scheduled to run on a pre-15.5 server and you need it to run on 15.5 servers too, leave the existing job alone and create a new job for 15.5 servers as the job commands are different.

Jobs intended to run on pre-15.5 servers do not need to be modified. The following table lists the templates that changed in version 3.0 and the changes you must make to jobs created from them.

Note: All templates listed below, except Delete Statistics, are incompatible with pre-15.0.1 servers. Do not use them to create jobs that are scheduled on pre-15.0.1 servers; you must use the 2.1 or 2.2 versions for pre-15.0.1 servers.

Table 12. Modified Job Scheduler Templates

Template	Modified file	Change description	Modify jobs created from this template
delete statistics	SybDeleteStatisticsTemplate.xml jst_delete_statistics	Added @ptn_name as third parameter.	Optional.
update statistics	SybUpdateStatisticsTemplate.xml jst_update_statistics	Added @ptn_name as fifth parameter and @datachg_threshold as the 10th parameter and added reference	Required, modify job SQL to include values (or NULL) for new parameters.
rebuild indexes	SybRebuildIndexTemplate.xml jst_reorg_rebuild_indexes	Added @ndx_ptn_name as third parameter.	Required, modify job SQL to include value (or NULL) for new parameter.
reclaim indexes	SybReclaimIndexTemplate.xml jst_reclaim_index_space	Added @ptn_name as third parameter	Required, modify job SQL to include value (or NULL) for new parameter.
reclaim tables	SybReclaimTableTemplate.xml jst_reclaim_table_space	Added @ptn_name as second parameter	Required, modify job SQL to include value (or NULL) for new parameter.
multiple	jst_get_free_space, jst_get_use_dspace	Replace reserved_pgs and data_pgs with reserved_pages and data_pages	No impact on job SQL.

7. Install the template stored procedures and and move them to the Job Scheduler template stored procedure directory. For example:

```
cd $SYBASE/$SYBASE_ASE/jobscheduler/Templates/sprocs
```

- a) Run the stored procedure install script for each server being upgraded:

```
installTemplateProcs <servername> <username> <password>
```

Note: Template stored procedures should be upgraded on all Job Scheduler servers and target servers that are upgraded to Adaptive Server version 15.5. Do not install them on pre-15.5 servers.

8. Install the template XML documents. Move to the JS template XML directory. For example:

```
cd $SYBASE/$SYBASE_ASE/jobscheduler/Templates/xml
```

- a) Run the XML install script on 15.0.1 servers with Job Scheduler installed:

```
installTemplateXml <servername> <machinename> <serverport>  
<username> <password> [<language_code>]
```

Use “en” for the *language_code* or omit the parameter completely as “en” is the default.

Note: Template XML should be upgraded on all Job Scheduler servers upgraded to Adaptive Server version 15.0.1. Do not install them on pre-15.0.1 servers or servers where Job Scheduler is not installed.

Upgrading High Availability and Cluster Support

There are two ways to upgrade the cluster subsystem.

This Adaptive Server release supports the following cluster platforms for high availability:

- HP-UX – MCSG 11.17
 - HPIA – MCSG 11.18
 - IBM AIX – HACMP 5.4
 - Sun Solaris – VCS4.0, SunCluster 3.2
 - Linux-AMD – VCS4.1
 - Win2003 – Cluster Manager 5.2
- Perform a major upgrade, which involves cluster downtime, then shutting down and restarting all servers. In this case, you must:
- a) Run **sp_companion suspend**, as described in *Using Sybase Failover in a High Availability System*.
 - b) Offline the resource groups of primary and secondary companions on both nodes. Ensure that the companion server and corresponding resource groups are not automatically brought online until the cluster system upgrade is complete.
 - c) Upgrade the cluster subsystem following the instructions from the cluster system vendor. You may find options to migrate the current resource groups to the new cluster version. If such an option is not available (or if the resource groups get deleted or corrupted), re-create the resource groups and configure them appropriately after you have upgraded the cluster system.
 - d) Bring the resource groups online. This should bring primary and secondary companions online on their respective nodes.

- e) Run **sp_companion resume** as described in *Using Sybase Failover in a High Availability System*.
- Perform a minor upgrade, to avoid cluster downtime. Nodes are failed over to other nodes and upgraded one at a time. For example, assuming ASE1 is the primary companion on node N1, and ASE2 is secondary companion running on node N2:
 - a) Upgrade the primary companion:
 1. Failover ASE1 from N1 to N2. Achieve this by relocating the primary resource group to N2 or by shutting down ASE1.
 2. Upgrade the cluster subsystem on N1 following the upgrade instructions provided by the vendor.
 3. Fail back ASE1 from N2 to N1. See the appropriate cluster chapter in *Using Sybase Failover in a High Availability System* for more details on Adaptive Server failback.
 - b) Upgrade the secondary companion. If you are using a symmetric configuration, follow the steps in “upgrade the primary companion,” above, for ASE2. If you are using an asymmetric configuration:
 1. Take the secondary resource group offline and verify ASE2 is shut down. ASE2 is not available during this upgrade.
 2. Upgrade the cluster subsystem on N2 following the upgrade instructions from the vendor.
 3. Start ASE2 by bringing the secondary resource group online to N2.

Migrating from 32-bit to 64-bit Versions on Different Computers

To migrate Adaptive Server from the 32-bit to the 64-bit version, first install and configure the 64-bit operating system.

There are three ways to migrate data on different computers:

- Using dump and load
- Using BCP
- Replacing the binary

After upgrading or migrating to other platforms, the server internally rebuilds procedures using the text in `syscomments` as it is first executed.

`syscomments` may not execute properly if it has code that drops existing objects and recreates them. Manually drop objects prior to executing `syscomments` for the first time.

Migrating Data Using a Dump and Load Method

Learn how to migrate data using a dump and load method to migrate data from a 32-bit to a 64-bit Adaptive Server version.

1. Run **dbcc** checks on all databases in the 32-bit Adaptive Server and make sure they run cleanly.
2. Create a 64-bit Adaptive Server in a new directory.
3. Create devices and databases to match those in the 32-bit server. Make sure that the *sysusages* mapping is correct.

Note: Allow 10 percent additional space for the *sysystemprocs* database.

4. Dump the databases on the 32-bit server.
5. Load the databases to the 64-bit server.
6. If you have partitioned tables, update partition statistics.
7. Run **dbcc** checks on the 64-bit server to ensure they run cleanly.

For information on upgrading compiled objects, see *Upgrading Compiled Objects With dbcc upgrade_object* on page 105.

Using BCP to Migrate Data

If you use DDL scripts to create devices, databases, tables, rules, stored procedures, triggers, and views, you can use **bcp** to migrate data from a 32-bit Adaptive Server to a 64-bit Adaptive Server.

If you do not have DDL scripts, use the **ddlgen** utility to re-create the schema for the Adaptive Server you are about to upgrade. See the *Utility Guide*.

1. Run **dbcc** checks on databases to verify data integrity.
2. Use **bcp** to extract all the data from all the tables in the databases.
3. Create a new 64-bit Adaptive Server in a new directory.
4. Create devices, databases, and tables.
5. Use **bcp** to bulk copy data into tables.
6. Re-create all views, triggers, stored procedures.
7. Run **dbcc** checks on the 64-bit server and make sure they run cleanly.

Migrating Data by Replacing the Binary

You can migrate data from a 32-bit server to a 64-bit server by replacing the binary.

1. Run **dbcc** checks (**checkdb**, **checkalloc**, **checkcatalog** and **checkstorage**) on all databases in the 32-bit Adaptive Server to ensure they are error free.
2. Copy the files for the 64-bit Adaptive Server into a new directory.

3. Shut down the 32-bit server.
4. Copy the `interfaces` file and the configuration file from the 32-bit `$SYBASE` directory to the 64-bit `$SYBASE` directory.
5. Copy the 32-bit `$SYBASE/$SYBASE_ASE/install/RUN_<server>` file to the equivalent 64-bit `$SYBASE/$SYBASE_ASE/install` directory.
6. Edit the `RUN_server` file to reflect the new location of the `interfaces`, configuration, and log files.
7. Remove all references to the 32-bit `$SYBASE` directories from your `$PATH` definition.
8. Change to the 64-bit `$SYBASE` directory and source the `SYBASE.csh` script (C shell).
9. Change to the 64-bit `$SYBASE/$SYBASE_ASE/install` directory and execute:


```
startserver -f RUN_server
```
10. After the 64-bit server starts, run `installmaster`, `installmodel`, and `instmsgs.ebf`.
11. If you used `dbccdb` for **dbcc checkstorage**, run `installdbccdb`. This re-creates tables in the `dbccdb`. This may cause you to lose data.
12. Drop and re-create the compiled objects, such as stored procedures, triggers, views, and defaults.
13. If you have partitioned tables, update partition statistics.
14. Run **dbcc** again on all databases to verify that they run cleanly.

Upgrading Using `allrows_dss` In Parallel

If you are using the configuration option `allrows_dss` when upgrading from a 15.0.x version to a 15.0.2 version in parallel mode, the server may stack trace when the optimizer selects a plan that requires `tempdb`, which has not yet been online.

Change configuration from `allrows_dss` when rebooting with the new binary.

After a minor upgrade is complete, you can return the server to its earlier configuration.

Recovering From a Failed Upgrade

Follow these steps after you have fixed the underlying problem that caused the failed upgrade.

1. Rerun the upgrade utility, `$SYBASE/$SYBASE_ASE/upgrade/upgrade .`
2. Reinstall system stored procedures using the scripts from the new version:

```
isql -Usa -P<password> -i $SYBASE/$SYBASE_ASE/scripts/
installmaster
```

Note: When the server crashes during an upgrade, as it is restarting it attempts to upgrade the databases. After this finishes, and the installation is fully recovered, re-run the upgrade

utility. This assures that all postupgrade configuration work is done. Then reinstall system stored procedures as previously described.

3. During the upgrade process, the log may become too full due to recorded catalog changes. If so, log in to the new server using **isql**, and issue:

```
isql> dump tran dbname with truncate_only
```

- a) If **truncate_only** indicates that it could not truncate the log, retry the command as **no_log** instead.
- b) If this too does not truncate the log successfully, then use the **alter database** command to extend the database as explained below.

This frees the log, and allows the upgrade process to continue.

4. In some cases, the pre-upgrade space estimations are insufficient for the data copy phase of upgrade. In this case, an error message says that there is insufficient space in the *system* segment for the upgrade. The upgrade process hangs, waiting for space to be provided. You can log in to the new server using **isql**, and use **alter database** to increase the size of the database.

```
isql> alter database dbname on device_name = "2m"
```

alter database allows you to specify the size to alter the database with the unit-specifier 'm', or 'M'.

Downgrading from Adaptive Server 15.5

Learn how to downgrade from Adaptive Server 15.5 to an earlier Adaptive Server 15.0, 15.0.1, 15.0.2 or 15.0.3 release. Downgrading to a 12.5.x or earlier release is not supported.

Note: Downgrade is not supported on the Linux Itanium platform.

Note: Sybase recommends that you do not use the new features offered in Adaptive Server version 15.5 until you are committed to staying with 15.5.

Additional actions are required if you used 15.5 features. Review the steps in the Additional Steps Required if you Used New Features topic.

If you used replicated databases with your server, complete additional steps in order to downgrade your server. See the Downgrading Adaptive Server with Replicated Databases topic.

A procedure called **sp_downgrade** is used to help with the downgrade process. This procedure requires 'sybase_ts_role' to be on. The 'sa_role' or 'sso_role' permissions are expected when running this procedure. For more detailed information on this system procedure and its syntax see 'The New System Procedures' section in the 'System Changes' chapter of the *Adaptive Server New Feature Guide*.

Note: If you used new features in Adaptive Server 15.5, returning to 15.0.2 ESD #3 is preferable, because this version can handle a number of error conditions that may arise as a result of downgrading.

If you return to a pre-15.0.2 ESD #3 version of Adaptive Server, no error handling is provided. Using the new features can result in inexplicable behavior, misleading error messages and stack traces unless you are uninstalling all of the usages of the new features reported by **sp_downgrade**.

1. Save the current auditing settings for the 15.5 Adaptive Server by saving the output of the following command in all system and user databases where any auditing options have been turned on:

```
1> sp_displayaudit
2> go
```

2. Back up all of your databases and the \$SYBASE release area prior to the 15.5 downgrade.
3. Start the 15.5 server in single user mode using **dataserver -m** option to guarantee that no other user can access Adaptive Server while you are going through the downgrade steps. See the *Utility Guide* for more details on starting the server in single user mode.
4. Make sure Adaptive Server 15.5 is prepared for downgrade by executing:

```
sp_downgrade 'prepare', @toversion='version'
```

Downgrading from Adaptive Server 15.5

The value of “version” may be written as “15.0”, “150”, “15.0.1”, “1501”, “15.0.2”, “1502”, “15.0.3” or “1503.” However, the version entered should be the Adaptive Server version to which you want to downgrade.

Note: If you have used the 15.0.2 features for encrypted columns and

- You are reverting back to Adaptive Server 15.0 GA or Adaptive Server 15.0 ESD #1 you must specify the version as '15.0.'
 - You are reverting back to Adaptive Server 15.0, 15.0 ESD #2, 15.0.1, or a 15.0.1 ESD you must specify the version parameter as '15.0.1.'
 - You are reverting back to Adaptive Server 15.0.2 or a 15.0.2 ESD, you must specify the version parameter as '15.0.2.'
-

sp_downgrade must be executed under master database.

sp_downgrade 'prepare' validates the readiness of Adaptive Server 15.5 for downgrade. This means that **sp_downgrade 'prepare'** may print messages regarding manual changes required before the downgrade can be completed. Errors reported must be corrected and implications of warnings reported must be understood before you can proceed with further downgrade steps.

5. Repeat the last step until **sp_downgrade 'prepare'** reports no errors.

6. Begin the downgrade process by executing:

```
sp_downgrade 'downgrade', @toversion='version',@override=1
```

The version number should be the same as the version you entered above. Once the above command completes successfully there can be no more activity on the 15.5 server. You must run checkpoint and immediately shutdown the 15.5 server.

7. Copy the RUN_SERVER file to the 15.0, 15.0.1, 15.0.2 or 15.0.3 release area you are downgrading to. Modify the RUN_SERVER file to use the dataserver binary of the version to which you are downgrading. For example, if you are downgrading to 15.0.2 update 15.5 related information to 15.0.2 related information. From the SYBASE directory:

```
/work_dirs/sybase/ase1502/ASE-15_0/bin/dataserver\  
-s OLD_SERVER_NAME\  
-d DOWNGRADED_MASTER_DEVICE\  
-e OLD_SERVER_LOG_FILE\  
-c OLD_SERVER_CONFIG_FILE\  
-M /work_dirs/sybase/ase1502\  

```

- Change the dataserver binary.
- You can use **-e**, **-c**, **-M** option point to the old server.

8. Restart the old server using the modified RUN_SERVER file. Ensure that the environment variable points to the old server release such as \$SYBASE, \$SYBASE_ASE, or \$SYBASE_OCS

Note: When downgrading to a 15.0 or a 15.0.1 server, user passwords may be reset depending on the value of **sp_passwordpolicy 'allow password downgrade'**. If **'allow password downgrade'** is set to '0', user passwords are reset and printed on the

console. Be sure to save the output of `sp_downgrade 'downgrade', <version>` into a file. So the passwords are not lost. The old password do not work. Otherwise, you must restart the downgraded server with `-psa` to regenerate the sa password.

9. Run any scripts that may be applicable to your site listed in the post-downgrade tasks.

These are the tasks needed for downgrade to Adaptive Server 15.0, 15.0.1, 15.0.2, 15.0.3 if you have not used any 15.5 functionality. If you have used 15.5 functionality then follow the additional steps described in the rest of the section.

Post-downgrade Tasks

After you perform the basic downgrade steps, some changes from the 15.5 upgrade still persist.

1. If you have both a local and a remote server running on an Adaptive Server version 15.x and are upgrading both of them, upgrade the local server first. If one is to be upgraded but not the other, upgrade the local server first.

Sybase does not certify that Component Integration Services running on earlier versions of Adaptive Server can connect to later versions. You may encounter errors if an earlier version of Adaptive Server maps a proxy table to a later version and the remote table uses functionality not available in the earlier release.

Sybase certifies that each version of Adaptive Server connects to earlier releases through Component Integration Services. Component Integration Services is tested and certified to connect to earlier versions of Adaptive Server.

2. Installing system stored procedures of the old version drops auditing information about the system stored procedures. Reenable auditing for all system stored procedures for which auditing was enabled before the downgrade. Use the output of `sp_displayaudit` recorded during predowngrade to help determine the system stored procedures for which auditing was enabled. Reenter the audit options using `sp_audit`. For example, if you had enabled stored procedure auditing for `sp_addlogin` stored procedure in your server before upgrading, run the following `sp_audit` command to reenable auditing on `sp_addlogin`:

```
sp_audit "exec_procedure", "all", "sp_addlogin", "on"
```

After you run `installmaster`, `installmodel`, `installcommit`, `installsecurity`, `installhasvss`, `installjsdb`, and `installmsgsvss` on your downgraded server, return system stored procedures to their earlier form. New stored procedures introduced in releases subsequent to the downgraded server are not removed.

Attention: Attempts to execute new stored procedures against an older binary have unpredictable results.

3. During upgrade to Adaptive Server 15.5 you should have created a `restore_msgs . ebf` file. Now you must run this script to restore your messages back to the version from which you upgraded.

```
isql -Usa -P <sa_password> -S <server_name> -i  
<restore_msgs . ebf>
```

Note: If you upgraded from a 15.0 or 15.0.1 release and are reverting back to the 15.0 or 15.0.1 release, then you must only run `instmsgs . ebf` file from the relevant release.

4. The on-disk structures of stored procedures, triggers, and views may contain statement identity tokens, datatypes and object references that are not understood by the earlier Adaptive Server versions. You must drop all compiled objects that use features introduced into Adaptive Server subsequent to the release to which you are downgrading.
5. If during the upgrade process you ran the ‘**update all statistics**’ on *syslogins* then you must **delete statistics** for *syslogins* and recreate them.

spt_values are dropped and re-created when you run **installmaster** from the release area you downgraded to. Any new types are eliminated from this table.

6. Running **installmaster** from the release area of the server to which you downgraded removes configuration parameters, that belong to 15.5, by deleting *sysconfigures* rows for configuration parameters that do not exist in *syscurconfigs*. After running **installmaster** the error messages no longer appear when you start the server.

If a downgraded server is booted using the 15.5 configuration file, the new options cause the message ‘Unknown parameter’ the unknown options are reported the first time the server is booted. These messages can be ignored because the configuration file is rewritten without the unknown options.

7. If you used keywords in 15.5, see the Additional Steps Required if You Used New Features topic for more information.
8. If you disabled replication before the downgrade process, enable it again. Consult the Reenabling Replication Server After the Dataserver Upgrade topic for more information.

Additional Steps Required if You Used New Features

If you used Adaptive Server 15.5 features that are not available in the Adaptive Server version to which you are downgrading, additional steps may be necessary before downgrading your server. In addition, some actions must be performed immediately after downgrading.

- Java in Adaptive Server – If user classes compiled by a Java version above 1.2 are installed, these classes do not work in pre-15.0.3 Adaptive Server versions. Earlier versions of Adaptive Server ignore the **enable pci** parameter. You can remove it. **sybpcidb** is not used by pre-15.0.3 Adaptive Server versions. This can be dropped if desired.
- SQL Replication – See Upgrading Servers with Replicated Databases,
- **sp_addserver** with `<filter>` parameter – If you used **sp_addserver** to create server entries with the **filter** parameter specified, you may have entries in

sys.servers.srvnetname column which are longer than 32 bytes. You must delete these entries before downgrade and add them back after downgrade.

In general, no additional steps are required when you are returning to an Adaptive Server version in which the feature was already available.

Downgrading Adaptive Server with Replicated Databases

When downgrading an Adaptive Server 15.5 that has replication enabled on databases, you must complete additional steps before starting the downgrade procedure.

If the server you are downgrading from has replicated databases and SQL replication has been enabled you must follow different procedures depending upon the version of Adaptive Server you are downgrading to.

If your replication system is using RepAgent and you are downgrading to a 15.0.2 ESD #3 or higher there are no additional steps required. Adaptive Server 15.0.2 ESD #3 can keep the data synchronized.

If you are not using RepAgent to replicate data, consult the specific documentation for the product you are using.

If you downgrade to a version lower than 15.0.2 ESD #3, Sybase recommends that you use the standard documented procedure to downgrade an Adaptive Server with replicated databases.

The standard procedure to downgrade Adaptive Server with replicated databases consist of ensuring that all replicated data in the primary database transaction log has been successfully transferred to the standby or replicate database. Once RepAgent is started again after the downgrade process, you must ensure that it does not read logs written with version 15.5.

1. Suspend transaction processing and replication activities
2. Drain the transaction logs for primary databases
3. Disable the secondary truncation point.

See the Upgrading Servers with Replicated Databases topic, for more detailed information.

Downgrading Job Scheduler

If you are downgrading to Adaptive Server versions 15.0 or 15.0.1, run **installjsdb** script from the lower version.

1. Disable Job Scheduler:

```
1> sp_configure "enable job scheduler", 0
2> go
1> sybmgmtdb..sp_sjobcontrol @name=NULL, @option="stop_js"
2> go
```

2. Follow the downgrade steps for your platform.
3. After downgrading run the **installjsdb** script from the lower version of Adaptive Server:

Downgrading from Adaptive Server 15.5

```
isql -Usa -Psa_password -Sservername  
-i$SYBASE/$SYBASE_ASE/scripts/installjsdb
```

Note: The directory with the location of the **isql** executable (\$SYBASE_OCS/bin) must be in your path.

4. Enable Job Scheduler:

```
sp_configure "enable job scheduler", 1
```

5. To start Job Scheduler, either restart the server, or enter:

```
1> use sybmgmtdb  
2> go  
1> sp_sjobcontrol @name=NULL, @option="start_js"  
2> go
```

If You Downgraded Back to the Original Installation After Upgrading to 15.5

Look here to learn what to do if you upgraded to Adaptive Server 15.5, then downgraded back to the original version.

Restore the original messages by running the following script after downgrading the server.

```
isql -Usa -P<sa-password> -irestoremsgs.ebf
```

Troubleshoot SySAM

If SySAM errors are not addressed in a timely fashion, Adaptive Server may cease to function. Learn about the SySAM grace period [here](#).

There may be more recent information published in the *SySAM Asset Management Users Guide* online at *SySAM FAQ*. For the most up-to-date troubleshooting information in this Installation Guide, go to the Sybase Product Manuals site *Product Manuals site*. If your Sybase product cannot find a required license, it continues to operate in grace mode, if a grace period is available. Determine the license status of your product using the output from **sp_lmconfig** command. The `Status` column shows `graced` for any license that is operating within a grace period.

Typically, you have a 30-day grace period to resolve any SySAM-related issues. When a license is awarded under grace, the Sybase product error log entry and optional e-mail message include the date and time when the grace period expires. In addition, you can execute **sp_lmconfig** to determine the expiration date. Your Sybase product disables the functionality corresponding to the license if the issue is not fixed by the expiration date. Adaptive Server shuts down an expired license if `ASE_CORE`.

The Sybase product log includes information about licenses awarded under grace mode. If e-mail notifications are set up for a severity of “warning,” e-mail messages are sent for this event. See the [Enabling and changing e-mail notifications](#) topic for more information. The warnings in the Adaptive Server error log and e-mail messages are repeated while the license is under grace mode.

Where to Look For SySAM Related Errors

All SySAM-related errors are printed in your Sybase product error log and are prefixed with "kernel SySAM." For example:

```
<timestamp> kernel SySAM: Failed to obtain 1 license(s)
for ASE_CORE feature from license file(s) or server(s).
```

See the [Error Log Locations](#) topic for where to find error logs.

If e-mail alerts are configured, recipients receive an e-mail whenever a SySAM event occurs.

You can always check the current state of SySAM licenses in your Sybase product using the **sp_lmconfig** command. The output shows the current SySAM configuration and the status of each license.

If you are using a served license model, the license server’s debug log, located in the `log` directory, provides information about any license checkout issues.

Troubleshooting SySAM

Learn methods for troubleshooting SySAM here.

Table 13. Troubleshooting SySAM

Description	What to do
<p>Cannot find license type during the Adaptive Server installation</p>	<p>The Adaptive Server installation program displays only the most commonly used license types. If you are using a legacy license or a new license type, your license may not be listed by the installation program.</p> <p>If you do not find the license type you need, select “Unknown” or “Other” from the drop-down list and complete the installation. After the installation is complete, use sp_lmconfig to change the license type. See the <i>Reference Manual:Procedures</i> for sp_lmconfig usage information.</p> <hr/> <p>Note: If you set the license type value to “Unknown” or “Other,” Adaptive Server uses the first available license. If you have licenses for more than one type, Adaptive Server may pick the incorrect license. Set the correct license type when you have multiple license types on the license server.</p>

Description	What to do
<p>The installation program produces a warning about not finding a suitable license</p>	<p>If you select a served license model, the install program checks for a license with the selected edition and license type. A warning message is displayed if no suitable license is available.</p> <p>If you are performing a new Adaptive Server installation or upgrading an existing Adaptive Server version 12.5.x or earlier, proceed with the installation. Adaptive Server provides a 30 day grace period to resolve licensing issues. Once the installation is complete, debug the license check-out failure. For more information, see “Adaptive Server cannot check out a license and starts with graced license instead” in this table.</p> <p>If you are upgrading an existing Adaptive Server version 15.0 or later, Adaptive Server may not start after the upgrade if you do not have a suitable license. Exit the installation program and see if the existing Adaptive Server is able to check-out licenses. If it is not, troubleshoot this issue and then proceed with the upgrade.</p> <p>If the existing Adaptive Server is able to check-out licenses, verify the license you are using authorizes use of the update you are applying. This can be determined using the date based version of your license file and the date this update was released. You must have a license with a date based version that is greater than or equal to the date the update was released. Adaptive Server provides a grace period for updating the licenses after support is renewed.</p> <p>The date based version of the license can be determined from the <i>Version</i> column of the output from sp_lmconfig, by looking at the license checkout message in the Adaptive Server error log, or by examining the license itself. For more information, see Appendix A, “Anatomy of a License,” of the <i>Sybase Software Asset Management Users Guide</i>.</p> <p>The release date of the Adaptive Server can be determined by the date displayed in the version string, cover letter, or SPDC download link. This data is also shown in the error message posted.</p> <p>You can download an updated license from SPDC, if your support was paid for the date the update was released. If you support was not paid, you are not authorized to install an update.</p> <p>See Appendix B of <i>Sybase Software Asset Management Users Guide</i> for explanation of how SySAM uses date-based versioning. See “Updating your Sybase licenses” in Chapter 3, “Getting and Using your Licenses,” in the <i>Sybase Software Asset Management Users Guide</i> for how to update your license.</p>

Description	What to do
<p>The license server executables and scripts are not installed</p>	<p>The <code>sysam</code> wrapper script and the executables for the license server (<code>lmgrd</code> and <code>SYBASE</code>) are not installed or found in the <code>SYSAM-2_0 bin</code> directory.</p> <p>The license server component is not selected for installation by default. To install the license server, choose the Custom install option and select license server component.</p> <p>Alternatively, you can download a standalone SySAM license server install program from SPDC.</p>
<p>The license server does not start</p>	<p>The most common cause for the license server failing to start is that there are no served licenses in the server. In order for license server to start, there needs to be at least one served license in the <code>licenses</code> directory. When you install a new license server, there are no served licenses in the licenses directory and the license server does not start until you activate a license from SPDC and copy it into the licenses directory.</p> <p>Other causes for license server not to start include:</p> <ul style="list-style-type: none"> • The license activated from SPDC was an unserved license. Examine the license files. Served licenses always start with a <code>SERVER</code> header. If you do not see a line starting with <code>SERVER</code> at the top of the license, you are using an unserved license which does not work with the license server. • If you are using a specific port number for the license, that port number may already be in use. Use <code>netstat -a</code> to verify that the port number is free. If not, free the port or use a different port number for the license server. • The host name recorded in the <code>SERVER</code> header does not match the actual host name. The host name is recorded next to the <code>SERVER</code> keyword. If it does not match, correct the host name in the license file or set it to <code>"this_host"</code>, which is a keyword that works with any host name. • The licenses copied may be activated for a different machine. Examine the Host ID recorded in the line. It is recorded next to the host name in the <code>SERVER</code> header. It must match the Host ID for the machine where license server will be running.

Description	What to do
<p>The license server does not recognize the license files</p>	<p>If the license server refuses to serve the licenses, this may be because:</p> <ul style="list-style-type: none"> • The license was activated for a different machine or activated using incorrect host ID. Examine the host ID recorded in the license with the host ID for the machine. If they are mismatched, check-in and regenerate the license with the correct host ID. • The license has been altered. Nothing in the license except some designated information can be altered. The license is rejected if you change other information. If the license is being created by typing from a paper copy, verify if there are any errors entering the license. You can also download a new copy of the activated license from SPDC. <p>If the Host ID for your platform is based on the Network adapter identification, verify that you have used an ID associated with a valid NIC. IDs associated with loopback adapters are not valid. If the ID associated with a removable network adapter was used, ensure that the adapter is attached to the computer.</p>
<p>Adaptive Server does not start, and generates a license check-out failure error</p>	<p>If Adaptive Server cannot check out the required license, it determines whether the license can be issued in grace mode. If a grace period cannot be given, the license is not granted. If the base license for Adaptive Server (ASE_CORE) was in grace and the issue is not fixed by the end of the grace period, Adaptive Server fails to start.</p> <p>For more information, see “Adaptive Server cannot check out a license and starts with graced license instead” in this table. The troubleshooting tips for resolving licenses issued under grace apply to solve this issue as well.</p>

Description	What to do
<p>Adaptive Server cannot check out a license and starts with graced license instead</p>	<p>Execute the sysam diag feature_name command from the SYSAM-2_0 bin directory, where <i>feature_name</i> is the SySAM feature name for Adaptive Server or the optional feature that is in grace mode. The feature name is printed in the Adaptive Server error log and in the optional e-mail notifications.</p> <p>If the diag command shows no licenses available for check-out, this may be due to one of the following reasons:</p> <ul style="list-style-type: none"> • If you are using a serviced model: <ul style="list-style-type: none"> • Verify that the license server is running and reachable from the machine you are using. Use "sysam status" to verify that you can access the license server. If you cannot access the license server, see if the license server is running. • If the license server is running, use "sysam status -f feature_name" to determine whether the license server is serving a license for the given feature. If not, obtain the correct licenses from SPDC. • All licenses on the license server may be in use. If "sysam status -f feature_name" indicates that no licenses are available, obtain additional licenses or shutdown existing instances of Adaptive Server. • If you are using an unserved model: <ul style="list-style-type: none"> • The license for the given feature may not be copied in the local licenses directory. Obtain the correct licenses from SPDC and copy them to the local licenses directory. • The license may have been activated for a different machine or activated with incorrect host ID. Check-in and reactivate the license with the correct host ID. • The available licenses are for a different operating system or architecture. Obtain license for the correct platform from SPDC. • If you are running in a terminal server environment, unserved licenses cannot be used. Set up served licenses for a terminal server environment. • The available license may not authorize use of this version of Adaptive Server. For information about date based versioning, see Appendix B, "Installing Product Updates, EBFs, and Support Renewal" in the <i>Sybase Software Asset Management Users Guide</i>. For information about how to update your license, see the "Updating your Sybase licenses" section in Chapter 3, "Getting and Using your License," in the <i>Sybase Software Asset Management Users Guide</i>.

Description	What to do
<p>Adaptive Server cannot checkout a license and starts with graced license instead (continued)</p>	<p>If the diag command shows the license for the given feature is available for check-out, the reason that Adaptive Server cannot check-out the license could be:</p> <ul style="list-style-type: none"> • The product edition or license type does not match the requirement. If the edition and license type configuration parameters are set, Adaptive Server uses only the license that matches these settings. <ol style="list-style-type: none"> 1. Run sp_lmconfig to see the configured values. 2. Check the VENDOR_STRING attribute of the licenses to make sure a matching license exists. <p>If a matching license does not exist, obtain a suitable license from SPDC or adjust the Adaptive Server configuration.</p> <p>If Adaptive Server cannot start because the grace period expired, examine the <code>servername.properties</code> file located in the <code>ASE-15_0/sysam</code> directory under your SYBASE directory for values. These values are stored in two lines that start with <code>PE=</code> and <code>LT=</code>. These lines are present only if the configuration is set.</p> • Also see “Adaptive Server cannot find license for optional feature, even though the license exists” in this table for more information.
<p>Adaptive Server shows graced licenses even after the issue has been fixed</p>	<p>Adaptive Server performs license checks periodically, and the license status is not immediately updated. It is updated only after the next heartbeat cycle is completed. This may take a few hours.</p>
<p>Adaptive Server cannot find license for an optional feature, even though the license exists</p>	<p>Execute "<code>sysam diag feature_name</code>" to ensure that license for the optional feature exists and can be checked out from the machine where Adaptive Server is running. If the feature exists, but cannot be checked out from Adaptive Server, this could be because:</p> <ul style="list-style-type: none"> • The optional feature is for a different edition. • The Active or Standby flags do not match for base Adaptive Server and the optional feature. <p>Execute sp_lmconfig to determine the edition and active and standby configuration value. The edition and active and standby values are displayed in the <i>Property Name / Property Value</i> table. The edition value is displayed in the <i>PE</i> row and active and standby value is displayed in the <i>AS</i> row.</p> <p>Compare the values of <i>PE</i> and <i>AS</i> for the Adaptive Server with the same values recorded in the <i>VENDOR_STRING</i> and <i>ISSUER</i> fields of the optional feature license. They must match. Some features like <i>ASE_HA</i> are not offered with standby pricing, so there is no "<code>AS=A S</code>" flag in the license for this feature. This field is not considered when searching for a license without active and standby capability.</p>

Description	What to do
<p>Adaptive Server does not start with the expected edition or license type</p>	<p>If the edition and license type configuration parameters are not set, Adaptive Server uses the first available ASE_CORE license. If you have multiple ASE_CORE licenses with different editions and license types, which license is the first available depends on many factors such as license file directory sort order, available licenses on the license server, and so on.</p> <p>Because what edition and license type is granted cannot be guaranteed, Sybase recommends that you set the edition and license type configuration parameters. When these parameters are set, you are assured that Adaptive Server starts with that configuration. If a matching license were to be unavailable, Adaptive server starts under grace (if available) to allow you to resolve the licensing issue.</p>
<p>Adaptive Server does not pick the correct license</p>	<p>When Adaptive Server looks for a license, it does so in the following locations:</p> <ul style="list-style-type: none"> • Any saved values in the <code>registry</code> on Windows platforms, or the <code>.flexlmrc</code> file on UNIX platforms. Use <code>lmpath -status</code> to view the saved values. • The values set for SYBASE_LICENSE_FILE and LM_LICENSE_FILE environment variables. • All license files (any file with a <code>.lic</code> extension) in the licenses directory under the SYSAM-2_0 directory. <p>All of the above locations are searched in the order specified until a suitable license is found. When a directory is specified, license files in that directory are loaded in the directory sort order.</p> <p>The first license that matches the feature name, version, edition, and license type filter is used. This license may not be what you were expecting to use. Adjust the above locations to control this.</p>

Enabling and Changing E-mail Notifications

Use **sp_lmconfig** to configure e-mail notification. To view the current notification status and configuration, execute **sp_lmconfig** without any parameters.

To enable e-mail notifications, set the **smtp host**, **smtp port**, **email sender**, **email recipients**, and **email severity** properties.

To change the severity that triggers e-mails, set the **email severity** property to: "ERROR", "WARNING", "INFORMATIONAL" or "NONE".

To change the list of e-mail recipients, set the **email recipients** property to a comma-separated list of e-mail addresses. E-mail addresses can include anything that is capable of receiving SMTP messages.

Calling Sybase Technical Support

If you call Sybase Technical Support for SySAM-related issues, have this information available:

- The Adaptive Server error log.
 - `$SYBASE/ASE-15_0/install/<servername>.log`
- If Adaptive Server does not start, the server's properties file:
 - `$SYBASE/ASE-15_0/sysam/<servername>.properties`
servername must correspond to the name of the Adaptive Server that does not start.
- The license files saved in the `SYSAM-2_0/licenses` directory on the machine running Adaptive Server.
- The value of `SYBASE_LICENSE_FILE` and `LM_LICENSE_FILE` environment variables.
- Output from **lmutil lmpath -status** command. `lmutil` is located in the `bin` directory under the `SYSAM-2_0/bin` folder.
- If you are using a served license model:
 - The licenses saved in the `$SYBASE/SYSAM-2_0/licenses` directory of the license server.
 - The license server log file in the `$SYBASE/SYSAM-2_0/log` directory.

Troubleshoot the Server

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 the Error Log Locations topic for more information.

The table, Troubleshooting guidelines for UNIX platforms, 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 **srvbuild** unexpectedly quits, or if you cannot correct the problem, see the *Error Messages and Troubleshooting Guide*.

Table 14. Troubleshooting guidelines for UNIX platforms

Problem	Possible cause and solution
<p>The installation program cannot start Adaptive Server.</p>	<p>Failure to start Adaptive Server is generally caused by a lack of available RAM or disk space.</p> <p>Make sure you meet RAM requirements. If you have the required RAM, remove, then reinstall all applications to the hard drive and restart the installation process.</p> <p>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.</p> <p>Verify that you are logged in as System Administrator. You must log in as an administrator to start Adaptive Server.</p> <p>Shut down Monitor Server before restarting Adaptive Server.</p> <hr/> <p>Note: Adaptive Server may not start if it cannot find a SySAM license and if the grace period has expired. Check the Adaptive Server error log for the cause of the license checkout failure and fix the issues.</p>
<p>After upgrading Adaptive Server, you cannot use srvbuild.</p>	<p>After you begin upgrading a server, you may be unable to use the same srvbuild session for other tasks. Exit and restart srvbuild.</p>

Problem	Possible cause and solution
The installation program cannot connect to the upgraded Adaptive Server.	After you begin upgrading a server, you may be unable to use the same srvbuild session for other tasks. Exit and restart srvbuild .
The installation program detects reserved word conflicts.	See the Stopping Adaptive Server After a Failure topic.
The upgrade fails.	See the Recovering From a Failed Upgrade topic.

Logs for Installation Utilities

The information in the error logs helps to determine the reason and possible solution for an error message. Find the default error log locations for the installation, upgrade, and setup utilities here.

Table 15. Error log locations for installation utilities

Utility	Error log location
Install Anywhere	<code>\$SYBASE/log/ASE_Suite.log</code>
srvbuildres	<code>\$SYBASE/\$SYBASE_ASE/init/logs/srvbuildMMDD.NNN</code> Where: <ul style="list-style-type: none"> • <i>MM</i> is the month. • <i>DD</i> is the date. • <i>NNN</i> is a three-digit number identifying the srvbuild session.
sqlupgrades	<ul style="list-style-type: none"> • <code>\$SYBASE/\$SYBASE_ASE/init/logs/sqlupgradeMMDD.NNN</code> • <code>\$SYBASE/\$SYBASE_ASE/upgrade/upgrade.NNN</code> (a temporary file created by the upgrade process)

Utility	Error log location
Install Anywhere	<code>\$SYBASE/log/ASE_Suite.log.log</code>

Logs For Sybase Servers

The information in the error logs helps to determine the reason and possible solution for an error message. Find the default error log locations for each Sybase server here.

Table 16. Error log locations for Sybase servers for UNIX

Server	Default error log path and file name
Adaptive Server	\$SYBASE/\$SYBASE_ASE/install/servername.log
Backup Server	\$SYBASE/\$SYBASE_ASE/install/servername.log
Monitor Server	\$SYBASE/\$SYBASE_ASE/install/servername.log
XP Server	\$SYBASE/\$SYBASE_ASE/install/servername.log

Solutions to Common Installation Problems

If this section does not describe the problem you are experiencing, see the Error Messages and Troubleshooting Guide.

Table 17. Solutions to common installation problems

Problem	Possible cause and solution
Cannot use X-Windows	<p>If the setup and configuration utilities do not display correctly, you may have to adjust the resolution on your monitor.</p> <p>To change to a smaller font size, issue the following UNIX commands:</p> <pre>% cd \$SYBASE/ASE-15_0 % 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 % </pre> <p>The installation utilities will now use approximately 25 percent less window space.</p>

Problem	Possible cause and solution
Cannot eject the CD or DVD from the drive	<p>If you cannot eject the CD from the drive, check:</p> <ul style="list-style-type: none"> • 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. • For sybhelp processes. If these processes exist, kill them using the UNIX kill command.
DISPLAY environment variable not set correctly	<p>This error message means that the DISPLAY environment variable on the <i>remote</i> machine is not set correctly to display the installer to your <i>local</i> machine:</p> <pre>The DISPLAY environment variable is not set correctly.</pre> <p>To correct the problem, enter the following command at the UNIX prompt of the <i>remote</i> machine, where <i>host_name</i> is the name of the machine on which you want the installer to appear (that is, on your <i>local</i> machine):For C shell:</p> <pre>setenv DISPLAY host_name:0.0</pre> <p>For Bourne shell:</p> <pre>DISPLAY=host_name:0.0; export DISPLAY</pre>
Client not authorized to connect to server	<p>If you run the installer and you get this error message, it means the remote machine does not have permission to display the user interface on the local machine where you start working:</p> <pre>Xlib: connection to "host_name" refused by server Xlib: Client is not authorized to connect to Server xhost: unable to open display "host_name"</pre> <p>To correct the problem:</p> <ol style="list-style-type: none"> 1. Enter the following command at the UNIX prompt of your <i>local</i> machine, where <i>remote_machine</i> is the machine on which you are running the installer: <pre>xhost +remote_machine</pre> 2. Restart the installer.
Address already in use	<p>This message from srvbuild means you entered a port number that is already in use:</p> <pre>kernel: ninit: bind, Address already in use</pre> <p>To correct the problem, enter a different port number on the srvbuild window. The command netstat -a produces a list of port numbers in use.</p>

Problem	Possible cause and solution
Adaptive Server fails to start	<p>The shared memory of the operating system may not be set high enough.</p> <p>To correct the problem, see the Installing the Server topic, for instructions on adjusting the shared memory value. Restart the installation or upgrade process.</p>
The Installer fails to start	<p>If the installer does not start, restart it using the debug mode:</p> <ol style="list-style-type: none"> 1. Set the environment variable LAX_DEBUG to true before running the installer.
Cannot start XP Server	<p>You may receive the following message from XP Server when it is invoked by xp_cmdshell or some other extended stored procedure:</p> <pre>Msg 11018, Level 16, State 1: Procedure "xp_cmdshell", Line 2: XP Server must be up for ESP to execute. (return status = -6)</pre> <p>Verify that there is an XP Server entry in the Adaptive Server <i>sys.servers</i> table. If you created XP Server in a different srvbuild session than Adaptive Server and you did not specify a related Adaptive Server, srvbuild cannot update the <i>sys.servers</i> table. Verify that the XP server exists in the interfaces file or LDAP server.</p> <p>Use sp_addserver to add an entry to the <i>sys.servers</i> table.</p>
Troubleshooting Resource File Installations	<p>If you encounter problems during the build, configuration, or upgrade process while using the srvbuild[res], sqlloc[res], or sqlupgrade[res], it may be that these utilities did not allow enough time for Adaptive Server to shut down properly. Set the SYBSHUTWAIT environment variable to force the utilities to wait for Adaptive Server to shut down. For example, this command forces the utility to wait for two minutes to allow Adaptive Server to shut down before proceeding with the next task:</p> <pre>% setenv SYBSHUTWAIT 120</pre>

Problem	Possible cause and solution
Configuration Failed	<p>Check the location of the system shared libraries. The shared libraries <code>libXt.a</code> and <code>libX11.a</code> are normally stored in <code>/usr/lpp/X11/lib/R5</code>. The <code>libXm.a</code> library is normally located in either <code>/usr/lpp/X11/Motif1.2/lib</code> or <code>/usr/lpp/X11/R5/Motif1.2</code>.</p> <p>If these shared libraries are located in directories different from the above, set the <code>LIBPATH</code> environment variable to indicate these other directories.</p> <p>The shared libraries <code>libXt.so</code> and <code>libX11.so</code> are normally stored in <code>/usr/openwin/lib</code>. The shared library <code>libsocket.so</code> is normally located in <code>/usr/lib</code>.</p> <p>If the shared libraries on your operating system are located in directories other than those mentioned in the preceding paragraph, set the environment variable <code>LD_LIBRARY_PATH</code> to indicate the new locations.</p>

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
```

3. If the installation or upgrade session fails after you start Adaptive Server, try using Sybase Central to shut down the server. If Sybase Central cannot shut down the server, use the **shutdown** command.

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 the Error log Locations topic for Sybase servers for UNIX.

If the Installation Quits While You Are Configuring Adaptive Server

Learn what to do if the installation suddenly quits.

1. View the contents of the log file generated by Adaptive Server. For default error log locations, see Error log locations for Sybase servers for UNIX.
2. Take any suggested actions to correct the problem.

3. If the installation fails **after** the installation program has created any operating system files, such as the *master* device or system procedures device files, delete those files.
4. If the installation fails **after** the installation program starts the Adaptive Server that you are attempting to install, shut down that server.
5. Use Server Config to restart the configuration.

If Adaptive Server Fails the Preupgrade Eligibility Test

If the server is not eligible for an upgrade.

If Adaptive Server fails the pre-upgrade test, Server Configuration 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 failed the pre-upgrade eligibility test.

After you resolve any problems, shut down Adaptive Server and use **sqlupgrade** to complete the upgrade session.

Recovering from a Failed Upgrade

If the upgrade process fails, 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 Error log locations for installation utilities.

Note: Once you start Adaptive Server on the new version you cannot start Adaptive Server on the earlier version. Attempts to do so result in having to restore from backups.

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 **Server Config** but do not attempt to restart the upgrade session until you have restored the databases from backup. After restoration is complete, retry the upgrade.

If the Cause of the Failure Is Unknown

If the upgrade process fails again, and you cannot determine the cause of failure look here for the location of the error log file.

By default, the log file is located in `$SYBASE/$SYBASE_ASE/install/<servername>.log`.

Check the error log file to find out when and where the upgrade failed, and contact Sybase Technical Support.

Re-running the Upgrade

Failures while upgrading a server installation fall into one of two categories: failure to upgrade an individual database, or failure to complete configuration changes after all databases have been upgraded.

1. First correct the problem that caused the failure. Output from the upgrade utility should identify the problem.

The most common cause of failure is running out of some resource: space (either data or log), locks, auxiliary scan descriptors. You can add space to the database using the **alter database** command. Other resource failures may be correctable by changing the server's configuration via the **sp_configure** stored procedure.

2. If an upgrade failure leaves a database offline, and the failure can only be corrected by making data changes in the database, you can gain access to the failed database by using **isql** or a similar program to connect to the affected server as user "sa" and issuing this command:

```
dbcc traceon(990)
```

Note: This trace flag grants access only to user "sa". It is not sufficient to use an account having "sa_role". If you have disabled the "sa" login, you must re-enable it to get access using this method.

Having set this trace flag, user "sa" can now use the offline database and make the necessary changes to correct the upgrade failure.

To restart a server that has failed to upgrade, use the **online database** command:

```
online database <failed_db_name>
```

The server restarts that database's upgrade from the point of failure.

3. If the failure occurs after all databases have been upgraded, or if a failure somehow causes the upgrade utility to crash, you can re-run the utility manually. First diagnose and correct the failure, then run the upgrade utility:

```
$_SYBASE/$_SYBASE_ASE/upgrade/upgrade
```

When restarted in this way, the upgrade process says it is "verifying" the upgrade rather than "starting" it, but it makes all the same checks as for the original upgrade.

4. To verify that a database has upgraded successfully, check any database's upgrade status using the **online database** command. If any upgrade is required for a database, this command performs it. You may also use a procedure such as this to check all databases at a given installation:

```
declare @dbname varchar(255)
select @dbname = min(name)
from master..sysdatabases
while @dbname is not null
```



```
begin
online database @dbname
select @dbname = min(name)
from master..sysdatabases
where name > @dbname
end
```

Note: There are certain upgrade failures from which the server cannot recover. For example, attempts to upgrade system tables to version 15.x format are quite sensitive to failures at certain points during the required changes. If you encounter such a failure, restore the failed database from backup. To prevent the upgrade from failing again, correct the problem that caused the original failure **before** issuing the **online database** command for that database. These catastrophic failures are nearly always caused by running out of resources, as described above, which then causes a failure to undo the aborted transaction.

If the Cause of the Failure Is Known

If the error logs or messages clearly indicate the cause of failure, and you do not believe your databases were damaged, you can attempt to fix the problem and re-run the upgrade immediately.

1. Exit the Server Config program.
2. Exit the **sqlupgrade** program.
3. 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.

4. If needed, shut down Adaptive Server.

Shutting down the server enables the installation program to start the server and re-run the upgrade session.

5. Start Server Config from the Sybase program folder.
6. Start **sqlupgrade** again.
7. Select **Upgrade Adaptive Server**, and proceed with the upgrade.

Upgrading Compiled Objects with Dbcc Upgrade_Object

Adaptive Server version 11.9.3 introduced the process of upgrading compiled objects based on their source text.

The compiled objects are:

- Check constraints
- Defaults

- Rules
- Stored procedures (including extended stored procedures)
- Triggers
- Views

The source text for each compiled object is stored in the *syscomments* table, unless it has been manually deleted. When you upgrade the server, the existence of the source text in *syscomments* is verified during that process. However, the compiled objects are not actually upgraded until they are invoked.

For example, if you have a user-defined stored procedure named **list_proc**, the presence of source text for **list_proc** is verified when you upgrade to Adaptive Server 15.x or 15.5. The first time **list_proc** is invoked after the upgrade, Adaptive Server detects that the **list_proc** compiled object has not been upgraded. Adaptive Server recompiles **list_proc**, based on the source text in *syscomments*. The newly compiled object is then executed.

Upgraded objects retain the same object ID and permissions that they used before being upgraded.

Compiled objects for which the source text was hidden using **sp_hidetext** are upgraded in the same manner as objects for which the source text is not hidden. For information on **sp_hidetext**, see the *Reference Manual:Procedures*.

Note: If you are upgrading from 32-bit installations to use a 64-bit Adaptive Server, the size of each 64-bit compiled object in the *sysprocedures* table in each database increases by approximately 55 percent when the object is upgraded. The pre-upgrade process calculates the exact size. Increase your upgraded database size accordingly.

To ensure that compiled objects have been upgraded successfully **before** they are invoked, you can upgrade them manually using the **dbcc upgrade_object** command.

For details, see Finding compiled object errors before production.

Finding Compiled Object Errors Before Production

Changes made in earlier versions of Adaptive Server may cause compiled objects to work differently in version 12.5.x and later.

Use **dbcc upgrade_object** to find the following errors and potential problem areas that may require manual changes to achieve the correct behavior:

- Missing, truncated, or corrupted source text
- Temporary table references
- Reserved word errors
- Quoted identifier errors
- select * potential problem areas

After reviewing the errors and potential problem areas, and fixing those that need to be changed, you can use **dbcc upgrade_object** to upgrade compiled objects manually instead

of waiting for the server to upgrade the objects automatically. For details, see Using `dbcc upgrade_object`.

Compiled Object problem	Description	Solution
Missing, Truncated, or Corrupted Source Text	If the source text in <i>syscomments</i> was deleted, truncated, or otherwise corrupted, dbcc upgrade_object may report syntax errors.	<ul style="list-style-type: none"> If the source text was not hidden, use sp_helptext to verify the completeness of the source text. If truncation or other corruption has occurred, drop and re-create the compiled object.
Temporary Table References	If a compiled object such as a stored procedure or trigger refers to a temporary table (<code>#temp table_name</code>) that was created outside the body of the object, the upgrade fails, and dbcc upgrade_object returns an error.	To correct this error, create the temporary table exactly as expected by the compiled object, then execute dbcc upgrade_object again. You must not do this if the compiled object is upgraded automatically when it is invoked.
Reserved Word Errors	For example, suppose you load a database dump from Adaptive Server 11.5 into Adaptive Server 15.0 and the dump contains a stored procedure that uses the word “lock.” When you run dbcc upgrade_object on that stored procedure, the command returns an error because, although “lock” was not reserved in version 11.5, it became a reserved word in version 11.9.2. With this advance notice, you can change the stored procedure and any related tables before they are used in a production environment.	Either manually change the object name or use quotes around the object name, and issue the command set quoted identifiers on . Then drop and re-create the compiled object.

Quoted Identifier Errors

Quoted identifiers are not the same as literals enclosed in double quotes. The latter do not require you to perform any special action before the upgrade.

dbcc upgrade_object returns a quoted identifier error if:

- The compiled object was created in a pre-11.9.2 version with quoted identifiers active (**set quoted identifiers on**).
- Quoted identifiers are not active (**set quoted identifiers off**) in the current session.

For compiled objects created in version 11.9.2 or later, the upgrade process automatically activates or deactivates quoted identifiers as appropriate.

1. To avoid this error, activate quoted identifiers before running **dbcc upgrade_object**.
When quoted identifiers are active, use single quotes instead of double quotes around quoted **dbcc upgrade_object** keywords.
2. If quoted identifier errors occur, use the **set** command to activate **quoted identifiers**, and then run **dbcc upgrade_object** to upgrade the object.

Select * Potential Problem Areas

In Adaptive Server version 11.9.3 and later, the result of a **select *** clause in a stored procedure, trigger, or view that was created in an earlier version of Adaptive Server may be different from what you expect.

For more information see the *Reference Manual*.

If **dbcc upgrade_object** finds a **select *** clause in the outermost query block of a stored procedure, it returns an error, and does not upgrade the object.

For example, consider these stored procedures:

```
create procedure myproc as
  select * from employees
go
```

```
create procedure yourproc as
  if exists (select * from employees)
    print "Found one!"
go
```

dbcc upgrade_object returns an error on **myproc** because **myproc** includes a statement with a **select *** clause in the outermost query block. This procedure is not upgraded.

dbcc upgrade_object does not return an error on **yourproc** because the **select *** clause occurs in a subquery. This procedure is upgraded.

Determining Whether Select * Should Be Changed in Views

If **dbcc upgrade_object** reports the existence of **select *** in a view, compare the output of *syscolumns* for the original view to the output of the table, to determine whether columns have been added to or deleted from the table since the view was created.

For example, suppose you have the following statement:

```
create view all_emps as select * from employees
```

Warning! Do not execute a **select *** statement from the view. Doing so upgrades the view and overwrites the information about the original column information in *syscolumns*.

Before upgrading the *all_emps* view, use the following queries to determine the number of columns in the original view and the number of columns in the updated table:

```
select name from syscolumns
  where id = object_id("all_emps")
select name from syscolumns
  where id = object_id("employees")
```

- a) Compare the output of the two queries by running **sp_help** on both the view and the tables that comprise the view.

This comparison works only for views, not for other compiled objects. To determine whether **select *** statements in other compiled objects need to be revised, review the source text of each compiled object.

If the table contains more columns than the view, then retaining the pre-upgrade results of the **select *** statement is important, change the **select *** statement to a **select** statement with specific column names.

- b) If the view was created from multiple tables, check the columns in all tables that comprise the view and rewrite the **select** statement if necessary.

Syntax for Using Dbcc Upgrade_object

Look here to learn about the parameters and syntax for using dbcc Upgrade.

Upgraded objects retain the same owner they had prior to being upgraded. Only the database owner or a System Administrator can execute **dbcc upgrade_object**. The database owner can upgrade their own objects in the database.

```
dbcc upgrade_object [ ( dbid | dbname
[, ['database.[owner].']compiled_object_name' |
'check' | 'default' | 'procedure' | 'rule' |
'trigger' | 'view'
[, 'force' ] ] ) ]
```

where:

- **dbid** – specifies the database ID. If you do not specify **dbid**, all compiled objects in the current database are upgraded.
- **dbname** – specifies the database name. If you do not specify **dbname**, all compiled objects in the current database are upgraded.
- **compiled_object_name** – is the name of a specific compiled object you want to upgrade. If you use the fully qualified name, **dbname** and **database** must match, and you must enclose the fully qualified name in quotes. If the database contains more than one compiled object of the same name, use the fully qualified name. Otherwise, all objects with the same name are parsed, and if no errors are found, upgraded.
- **check** – upgrades all check constraints and rules. Referential constraints are not compiled objects and do not require upgrading.

- **default** – upgrades all declarative defaults and the defaults created with the **create default** command.
- **procedure** – upgrades all stored procedures.
- **rule** – upgrades all rules and check constraints.
- **trigger** – upgrades all triggers.
- **view** – upgrades all views.

The keywords **check**, **default**, **procedure**, **rule**, **trigger**, and **view** specify the classes of compiled objects to be upgraded. When you specify a class, all objects in that class, in the specified database, are upgraded, provided that **dbcc upgrade_object** finds no errors or potential problem areas.

- **force** specifies that you want to upgrade the specified object even if it contains a **select *** clause. Do not use **force** unless you have confirmed that the **select *** statement does not return unexpected results. The **force** option does not upgrade objects that contain reserved words, contain truncated or missing source text, refer to nonexistent temporary tables, or do not match the quoted identifier setting. You must fix these objects before they can be upgraded.

Note: If **set quoted identifiers** is **on**, use single quotes around the keywords. If **set quoted identifiers** is **off**, you can use either double quotes or single quotes.

Syntax Examples for Upgrading Compiled Objects

Look here for examples of how to upgrade compiled objects.

Syntax	Description of what it does
<code>dbcc upgrade_object</code>	Upgrades all compiled objects in the active database
<code>dbcc upgrade_object(listdb, 'procedure')</code>	Upgrades all stored procedures in the <i>listdb</i> database. Single quotes are used around procedure because set quoted identifiers is on .
<code>dbcc upgrade_object(listdb, "rule")</code>	Upgrades all rules and check constraints in the <i>listdb</i> database. Double quotes are used around rule because set quoted identifiers is off .
<code>dbcc upgrade_object(listdb, list_proc)</code>	Upgrades all stored procedures named list_proc in the <i>listdb</i> database.
<code>dbcc upgrade_object(listdb, "listdb.jkarrik.list_proc")</code>	Upgrades the stored procedure list_proc , which is owned by the login "jkarrik".
<code>dbcc upgrade_object(master, "listdb.jkarrik.list_proc")</code>	Returns an error because the value of <i>dbname</i> is <i>master</i> and the value of <i>database</i> is <i>listdb</i> . These values must match.

Increasing the Log Segment Size

You can specify that all compiled objects of a particular class should be upgraded in one execution of **dbcc upgrade_object**; for example, you can upgrade all triggers by using the **trigger** keyword.

Even though you use only one **dbcc** command, the upgrade of each object is recorded in a separate transaction; the old row is deleted from *sysprocedures* and a new row is written. Therefore, if you run **dbcc upgrade_object** on a large number of compiled objects, your system may run out of log space.

1. Increase the size of the log segment in the databases in which you plan to run this command, to allow sufficient room to log all the upgrades.
2. To send all the output from **dbcc upgrade_object** to the window, execute **dbcc traceon(3604)**.

Sybase recommends that you use this command if you think the output of error messages might overflow the error log.

Upgrading Using Dump and Load

You can upgrade the databases and load pre-12.5 database dumps and transaction logs.

Some issues of which you should be aware:

- Upgrading requires space for copying data and logging changes to the system tables during the upgrade process. If the source database in the dump was full, the upgrade process might fail. While this is expected to be uncommon, you can use **alter database** to extend the free space in the event of insufficient-space errors.
- After reloading an older dump, run **sp_checkreswords** from the new installation on the loaded database to check for reserved words.

Upgrading Compiled Objects in Database Dumps

When you load a database dump that was created in an earlier version than the current Adaptive Server you are running, you are not required to perform the preupgrade tasks before loading the dump. Learn how to determine whether a compiled object has been upgraded.

You do not receive any notification if the compiled objects in your database dump are missing their source text. After loading a database dump, run **sp_checksource** to verify the existence of the source text for all compiled objects in the database. Then, you can allow the compiled objects to be upgraded as they are executed, or you can run **dbcc upgrade_object** to find potential problems and upgrade objects manually.

For information on using **sp_checksource**, see the *Reference Manual: Procedures*.

Troubleshoot the Server

1. Look at the *sysprocedures.version* column. If the object was upgraded, this column contains the number 12500.
2. If you are upgrading to a 64-bit pointer size in the same version, look at the *sysprocedures.status* column. It contains a hexadecimal bit setting of 0x2 to indicate that the object uses 64-bit pointers. If the bit is not set, the object is a 32-bit object, which means the object has not been upgraded.

Raw Partitions on Linux

You can create and mount database devices on raw bound devices for raw disk IO. Raw disk IO enables direct memory access from your address space to the physical sectors on the disk while omitting needless memory copy operations from the user address space to the kernel buffers.

Raw disk IO assumes that logical and physical IO are simultaneous, and writes are guaranteed to flush to the disk when the system **write()** call returns. When preparing a raw partition device, follow these guidelines:

1. Do not initialize a database device on the partition that contains your Sybase installation software. Doing so destroys all existing files on that partition.
2. 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.
3. 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 can be reused with the **disk resize** command.
4. To avoid any possibility of using a partition that contains the partition map, do not use cylinder 0.
5. Place the database device on a character device, because the server recovery system needs unbuffered system IO.
6. To determine whether a device is a block device or a character device run:

```
ls -l <device path>
```

Choosing a Raw Partition

Learn how 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.
4. Verify with the operating system administrator that the partition you have chosen is available.
5. Make sure the “sybase” user has read and write privileges to the raw partition.

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

Creating Raw Partitions

Specific system administration is required before raw devices can be enabled and used. The available tools to configure devices depend on the distribution configuration.

Physical disk space must be allocated in partitions on the disks where you want to set up raw devices. The physical IO subsystem can be on either SCSI or EIDE devices.

Note: You can create partitions with the Linux default **fdisk(8)** utility. You must have “root” privileges to use the command **fdisk**. See the **fdisk(8)** man pages for a complete description of the command.

1. **Example 1** - This example shows how to set up partitions as raw devices, on four SCSI disks in the system—*sda*, *sdb*, *sdc*, and *sdd*.

- a) Start *fdisk* on `/dev/sdd`:

```
# fdisk /dev/sdd
```

The system returns:

```
The number of cylinders for this disk is set to 8683
....
Command (m for help):
```

- b) Enter “p” to print the current partition layout. The output is:

```
Disk /dev/sdd: 64 heads, 32 sectors, 8683 cylinders
Units = cylinders of 2048 * 512 bytes
Device Boot Start      End    Blocks  Id System
/dev/sdd1      1    7499   7678960  83  Linux
/dev/sdd2     7500   8012    525312  82  Linux swap
/dev/sdd4     8013   8683    687104   5  Extended
```

2. **Example 2** - This example shows the extended partition (*sdd4*) has 687104 free blocks, starting from 8013 and ending at 8683. You can assign the remaining partitions later. This example assigns an additional partition for raw bound disk IO:

- a) Use the **n** command to create a new partition, and enter “l” at this prompt for “logical”.

```
Command (m for help):n
Command action
l  logical (5 or over)
p  primary partition (1-4)
```

- b) Accept the default by pressing Enter when you see:

```
First cylinder (8013-8683, default 8013):
```

- c) Accept the default by pressing Enter again, when you see:

```
Last cylinder or +size or +sizeM or +sizeK
(8013-8683, default 8683): 8269
```

- d) Use the **t** command, enter “5” at this prompt:

```
Partition number (1-8):5
```

- e) Enter “60” at this prompt:

```
Hex code (type L to list codes): 60
```

The output is:

```
Changed system type of partition 5 to 60 (Unknown)
```

- f) Repeat the above steps to create four partitions for raw device IO.
- g) Verify the setup using **p** to print the full partition table before writing it out. Make sure that there are no overlapping partitions and the type for the unassigned partitions is Unknown type 60.

The partition table can now be written to disk and you can quit the *fdisk(8)* utility.

Red Hat Raw Device Administration

Red Hat Enterprise Linux is fully equipped with the administration tools to set up raw devices and administrate them during a system startup. To set up a Red Hat Enterprise Linux system, the administrator partitions using the `/etc/sysconfig/rawdevices` file.

This is a plain text file that contains comments and examples for possible configurations:

```
# raw device bindings
# format: rawdev major minor
#         rawdev blockdev
# example: /dev/raw/raw1 /dev/sda1
#          /dev/raw/raw2 8 5
/dev/raw/raw1 /dev/sdd1
/dev/raw/raw2 /dev/sdd2
/dev/raw/raw3 /dev/sdd3
/dev/raw/raw4 /dev/sdd4
```

1. Once you have created them, bind the raw devices by starting them from `/etc/rc.d/init.d/rawdevices`.

```
[root@legolas init.d]# cd /etc/rc.d/init.d
[root@legolas init.d]# sh rawdevices start
Assigning devices:
/dev/raw/raw1 --> /dev/sdd5
/dev/raw/raw1:   bound to major 3, minor 5
/dev/raw/raw2 --> /dev/sdd6
/dev/raw/raw2:   bound to major 3, minor 6
/dev/raw/raw3 --> /dev/sdd7
/dev/raw/raw3:   bound to major 3, minor 7
/dev/raw/raw4 --> /dev/sdd8
/dev/raw/raw4:   bound to major 3, minor 8
done
```

2. To guarantee that the raw device binding occurs during any restart, use the *chkconfig(8)* utility.

```
# /sbin/chkconfig rawdevices on
```

SuSE Raw Device Administration

Administer raw disk partitions in the `/etc/raw` file, which is a plain text file containing comments and examples for possible configurations:

```
# /etc/raw
#
# sample configuration to bind raw devices
# to block devices
#
# The format of this file is:
# raw<N>:<blockdev>
#
# example:
# -----
# raw1:hdb1
#
# this means: bind /dev/raw/raw1 to /dev/hdb1
#
# ...
raw1:sda7
raw2:sda8
raw3:sda9
```

Once you have created them, bind the raw devices by starting them with the script `/etc/init.d/raw`:

```
# cd /etc/init.d
# sh raw start
bind /dev/raw/raw1 to /dev/sdb1... done
bind /dev/raw/raw2 to /dev/sdb2... done
bind /dev/raw/raw3 to /dev/sdb3... done
...
```

Use the **chkconfig(8)** utility to guarantee that the raw device binding occurs during any restart:

```
# /sbin/chkconfig raw on
```

Accessing Raw Devices from the Server

Once you have created the partitions and bound the devices for raw disk IO, Adaptive Server can use them. If Adaptive Server is running as user “sybase”, then apply read, write, and owner permissions to the `/dev/raw/raw#` device entries and the raw bound control device `/dev/rawctl`.

See the `chown(1)`, `chgrp(1)`, and `chmod(1)` commands to apply correct permissions.

1. To verify your settings, query the device binding using the **raw** command, enter:

```
# raw -qa
```

The output should be:

```
/dev/raw/raw1: bound to major 3, minor 5  
/dev/raw/raw2: bound to major 3, minor 6  
/dev/raw/raw3: bound to major 3, minor 7  
/dev/raw/raw4: bound to major 3, minor 8
```

2. You should have root privileges to execute this command. If you do not, you may see the following output for the `raw -qa` command:

```
Cannot open master raw device '/dev/rawctl'  
(Permission denied)
```

3. Using the raw devices, Adaptive Server and the installation and configuration utility, **srvbuild**, detects and presents size automatically. Enter the absolute path to the raw device when creating the *master*, *syssystemprocs*, *sybtempdb*, or any other device.

Upgrading Servers with Replicated Databases

After completing an upgrade, you cannot scan any part of the transaction log that existed prior to the upgrade, so you must follow the procedure outlined in this section if your server contains replicated primary databases (this includes replicated RSSD). This procedure helps to ensure that data from a replicated database has been successfully replicated to the replicate database.

Warning! You must have a valid **dump database** and a **dump transaction** before executing the procedures in the procedures for upgrading a replicated database.

The procedures described here do not upgrade Replication Server itself. For information on upgrading Replication Server, see the Replication Server documentation.

These steps precede the database upgrade procedure. Each step represents a topic in this section:

- Suspending transaction processing and replication activities
- Draining the transaction logs for primary databases
- Draining the RSSD transaction log
- Disabling the secondary truncation point

Determining If Existing Servers Contain Replicated Databases

Determine if existing servers contain replicated databases before upgrade.

1. Use **isql** to connect to the Server you are upgrading.
2. If your server uses Replication Agent™ to replicate data, connect to the server and execute:

```
1> sp_config_rep_agent
2> go
```

The above command displays the list of databases enabled for replication agent.

- a) If you use another application to replicate data, LTM or Mirror Activator, run the following command in each database including the master database:

```
1> dbcc gettrunc
2> go
```

If the command returns “1” for “*secondary trunc state*” in any database, replication is enabled in that database.

Suspending Transaction Processing and Replication Activities

You must suspend transaction processing and replication activities before upgrading a replicated database.

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

1. Before beginning the upgrade, verify that subscriptions created by the **create subscription** command, with primary data in the databases being upgraded, have reached the “valid” state at the primary Replication Server. Use the **check subscription** command to find the “valid” state.
2. Defer the upgrade process while the subscriptions are being created so that Replication Server does not attempt to access the database being upgraded.
3. Execute **rs_helproute** in each Replication Server System Database (RSSD) being upgraded.
4. The status of all existing routes should be “Active.” If any routes are not active, resolve them before continuing. See the Replication Server documentation for help in diagnosing and fixing the problem with the route, and then go on to the next step which is to shut down all applications that use the databases you are upgrading.
5. Shut down all applications that use the databases you are upgrading.
6. Use the **admin who** command in Replication Server to find the existing Data Server Interface (DSI) connections to the data server being upgraded.
7. 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
```
8. Leave the DSI connections to the RSSD databases running.

Draining the Transaction Logs for Primary Databases

Ensure that the Replication Server on each primary database you are upgrading completely processes the pre-upgrade log.

Follow these steps to ensure the transaction log is completely drained off.

1. Wait for all remaining transactions to be replicated.
2. Check that a manual update on a replicate table is forwarded to Replication Server:
 - a) If the database is enabled for Replication Agent:

```
1> use <dbname>
2> go
```



```
1> sp_config_rep_agent <dbname>, "traceon", "9201"
2> go
```

If you cannot access the console where Adaptive Server is running, you must also define a trace log file to check the information Replication Agent will send to it:

```
1> sp_config_rep_agent <dbname>, "trace_log_file", "path for
the log file"
2> go
```

- b) If the database uses a different method to replicate data, ensure that data is replicated by consulting the Replication Server queues. To do this, run the following Replication Server command:

```
1> admin who, sqm
2> go
```

3. Use **isql** to update one row in a single replicated table in the primary database:

```
1> update table set column = column
2> where key = unique_value
3> go
```

The **update** command helps track whether all modifications to the replicated database have been sent to the Replication Server.

Note: Sybase recommends you use a dummy table with a primary key which you define. It is recommended that you define a replication definition with all columns in the table.

4. In the primary Replication Server, execute the **admin who, sqm** command until the last *segment: block* entry for the inbound queue changes.
5. Execute the following Replication Server command to dump the last block of the inbound queue to the dump file you created earlier in this process:

```
1> sysadmin dump_queue, queue_number,
2> queue_type, last_seg, block, 1
3> go
```

6. Use Notepad or another text editor to examine the dump file to make sure it contains the transaction that corresponds to the update you performed above.
7. Repeat the update steps above until the transaction that corresponds to the update is in the dump file.
8. Stop the application or process reading the transaction log from the primary database. If you are using:

- a) Rep Agent – log into the Adaptive Server, and stop the Rep Agent:

```
1> sp_stop_rep_agent database
2> go
```

- b) LTM or Mirror Activator – stop them.

After draining the transaction logs, disallow all other activity in the databases. If activity does occur, you must retrain the logs.

9. Instruct Replication Server to reject incoming connections from Replication Agent, LTM or Mirror Activator by logging in to the Replication Server and suspending the Log Transfer connection from that database:

```
1> suspend log transfer from server.database  
2> go
```

Draining the RSSD Transaction Log

If the Replication Server has routes to other Replication Servers, ensure that the Replication Server processes all transactions in the RSSD transaction log before upgrading the databases.

To ensure that 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 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 in to 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 need not reference an actual table.

2. Log in to the replicate RSSD (not the primary RSSD) and execute the following query to find out if the replication definition has arrived from the primary RSSD:

```
1> select * from rs_objects  
2> where objname = "rep_def_name"  
3> go
```

If this **select** statement returns rows, the last replication definition created in step 1 has been sent successfully to the replicate RSSD. This means that the transaction log has been drained.

3. Log in to the replicate Replication Server and suspend the log transfer connection from the primary RSSD:

```
1> suspend log transfer from server.database  
2> go
```

4. If you are using Rep Agent, log in to the Adaptive Server, and stop the Rep Agent:

```
1> use database  
2> go
```

```
1> sp_stop_rep_agent database  
2> go
```

5. If you are using LTM, shut down the LTM.

6. If this is a replicated RSSD, log in to the Replication Server of the RSSD and issue:

```
1> sysadmin hibernate_on, 'Replication_Server_name'  
2> go
```

Disabling the Secondary Truncation Point

When upgrading a primary database, the Rep Agent, Log Transfer Agent or Mirror Activator must not be running, and you should turn off the secondary truncation point for the duration of the upgrade. The Rep Agent or Log Transfer Manager should be shut down prior to the upgrade.

For each primary database and replicated RSSD, disable the secondary truncation point:

1. Disable the secondary truncation point in the Adaptive Server database that is being upgraded by issuing:

```
1> use database
2> go
```

```
1> dbcc settrunc('ltm', 'ignore')
2> go
```

2. If the **dbcc settrunc** command fails, make sure that the Rep Agent or LTM are not running. When the Rep Agent and LTM are disabled, repeat the first step.
3. Truncate the primary transaction log running the command:

```
1> dump tran primary_dbname with truncate_only
2> go
```

4. Stop Replication Server then zero out the Replication truncation point. In the RSSD for the Replication Server, run:

```
1> rs_zeroltm primary_servername, primary_dbname
2> go
```


Using Sybssystemprocs

sysprocsdev is the default system name for this device. However, it is frequently referred to as the *sybssystemprocs* device, as in the Adaptive Server attribute window. This is because it stores the *sybssystemprocs* database.

Verify that the *sybssystemprocs* database is large enough. For an upgrade, the recommended minimum size for *sybssystemprocs* is the larger of 140MB, or enough free space to accommodate the existing *sybssystemprocs* database, and the largest catalog that is to be upgraded, plus an additional 10 percent of the largest catalog's size. The additional 10 percent is for logging upgrade changes.

You may need more space if you are adding user-defined stored procedures. 140MB accommodate additional internal data structures, but does not account for the possibility of a large number of user-defined system procedures.

If your *sybssystemprocs* database does not meet these requirements and you have enough room on the device to expand the database to the required size, use the **alter database** command to increase the database size.

Use **sp_helpdb** to determine the size of the *sybssystemprocs* database:

```
1> sp_helpdb sybssystemprocs
2> go
```

Use **sp_helpdevice** to determine the size of the *sysprocsdev* device:

```
1> sp_helpdevice sysprocdev
2> go
```

If the *db_size* setting is less than the required minimum, increase the size of *sysprocdev*.

Increasing the Size of the Sybssystemprocs Database

If your current *sybssystemprocs* database does not have the minimum space required, two options exist for creating a new database with sufficient space for the upgrade. Learn about these option here.

- The first option is to drop the old database and device and create a new *sysprocsdev* device.
- The second option is to leave the old database and device alone and add a new device large enough to hold the additional megabytes, and alter the *sybssystemprocs* onto the new device. Sybase recommends that you expand the current device.

To increase the size of the **sybssystemprocs** database:

1. If you do not have a current backup of your old database, create one now.
2. In **isql**, use **alter database** to increase the size of the *sybssystemprocs* database. For example:

```
1> use master
2> go
1> alter database sybssystemprocs on sysprocsdev=40
2> go
```

In this example, “sysprocsdev” is the logical name of the existing system procedures device, and “40” is the number of megabytes of space to add.

If the system procedures device is too small, you may receive a message similar to the following when you try to increase the size of the *sybssystemprocs* database:

```
Could not find enough space on disks to extend
database sybssystemprocs
```

a) If there is space available on another device, expand *sybssystemprocs* to a second device, or initialize another device that is large enough. See the [Increasing Device and Database Capacity for System Procedures](#) topic for more information.

3. To verify that Adaptive Server has allocated more space to *sybssystemprocs*, issue:

```
1> sp_helpdb sybssystemprocs
2> go
```

When the system procedures database is large enough to accommodate the increased size of the *sybssystemprocs* database, continue with the other pre-upgrade tasks.

Increasing Device and Database Capacity for System Procedures

If you cannot fit the enlarged *sybssystemprocs* database on the system procedures device, increase the size of the device and create a new database. Look here for how to do this.

This procedure involves dropping the database. For more information on **drop database**, see the *Reference Manual*.

Warning! This procedure removes all stored procedures you have created at your site. Before you begin, save your local stored procedures using the **defncopy** utility. See the *Utility Guide* for more information.

1. Determine which device or devices you must remove.

Warning! Do **not** remove any device that is in use by a database other than *sybssystemprocs*, or you will destroy that database.

```
select d.name, d.phyname
from sysdevices d, sysusages u
where u.vstart between d.low and d.high
and u.dbid = db_id("sybssystemprocs")
and d.status & 2 = 2
and not exists (select vstart
from sysusages u2
where u2.dbid != u.dbid
and u2.vstart between d.low and d.high)
```

- **d.name** – is the list of devices to remove from *sysdevices*.
- **d.phyname** – is the list of files to remove from your computer.

The **not exists** clause in this query excludes devices that are used by *sybssystemprocs* and other databases.

Make a note of the names of the devices to use in the following steps.

2. Drop *sybssystemprocs*:

```
1> use master
2> go
1> drop database sybssystemprocs
2> go
```

Note: In versions of Adaptive Server Enterprise earlier than 15.x, use **sysdevices** to determine which device has a low through high virtual page range that includes the *vstart* from step 2.

In version 15.x, select the *vdevno* from *sysusages* matching, the *dbid* retrieved in step 1.

3. Remove the device or devices using **sp_configure**:

```
1> sp_configure "allow updates", 1
2> go
1> delete sysdevices
    where name in ("devname1", "devname2", ...)
2> go
1> sp_configure "allow updates", 0
2> go
```

The **where** clause contains the list of device names returned by the query in step 1.

Note: Each device name must have quotes. For example, “devname1”, “devname2”, and so on.

If any of the named devices are OS files rather than raw partitions, use the appropriate OS commands to remove those files.

4. Remove all files for the list of **d.phyname** that were returned.

Note: File names cannot be complete path names. If you use relative paths, they are relative to the directory from which your server was started.

5. Find another existing device that meets the requirements for additional free space, or use a **disk init** command similar to the following to create an additional device for *sybssystemprocs* where */sybase/work/* is the full, absolute path to your system procedures device:

```
1> use master
2> go

1> disk init
2> name = "sysprocsdev",
3> physname = "/sybase/work/sysproc.dat",
```

Using Sybssystemprocs

```
4> size = 51200
5> go
```

Note: Server versions 12.0.x and later accept, but do not require "vdevno=number". In versions earlier than 12.0.x, 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 112MB (112 x 512 = 57344). For more information on **disk init**, see the *Reference Manual*.

6. Create a *sybssystemprocs* database of the appropriate size on that device, for example:

```
1> create database sybssystemprocs on sysprocsdev =
    112
```

```
2> go
```

7. Run the **installmaster** script in the old server installation directory. Enter:

```
isql -Usa -Ppassword -Sserver_name -i$SYBASE/ASE-15_0/scripts/
installmaster
```


Obtaining Help and Additional Information

Use the Sybase Getting Started CD, Product Manuals site (SyBooks™ Online), and online help to learn more about this product release.

- The Getting Started CD (or download) – contains release bulletins and installation guides in PDF format, and may contain other documents or updated information.
- Product Manuals at <http://sybooks.sybase.com/> – is an online version of Sybase documentation that you can access using a standard Web browser. You can browse documents online, or download them as PDFs. In addition to product documentation, the Web site also has links to EBFs/Maintenance, Technical Documents, Case Management, Solved Cases, Community Forums/Newsgroups, and other resources.
- Online help in the product, if available.

To read or print PDF documents, you need Adobe Acrobat Reader, which is available as a free download from the *Adobe* Web site.

Note: A more recent release bulletin, with critical product or document information added after the product release, may be available from the Product Manuals Web site.

Index

A

- Accessing SPDC
 - Sybase product download
 - online Web portal 10
- Adaptive Server Editions 1
- additional steps for downgrading 15.5 features 84
- applications
 - changing after renaming objects 56

B

- Backing up the server 40
- basic downgrade steps using sp_downgrade 81

C

- command line 37
- configuration parameters 83
- conflicting object names 57

D

- Determining the current edition 1
- determining whether select* should be changed in
 - views 108
- directory changes 44
- disabling
 - secondary truncation point 123
- downgrade 81
- downgrading back to the original installation after
 - upgrading 86
- downgrading job scheduler 85
- draining
 - primary databases 120
 - transaction logs 120
 - RSSD 122
- dsedit utility 67

E

- enabling
 - PCI 63
- Enabling the Java in the database in a high
 - availability system 71

- error log files
 - troubleshooting and 102

F

- fails preupgrade eligibility 103
- file descriptors 25
- finding compiled object errors before production
 - 106

H

- high availability 40
- Host ID
 - how to get
 - obtaining 28

I

- identifiers
 - conflicting with reserved words 56
- increasing
 - device and database capacity
 - system procedures 126
 - increasing the log segment size 111
- Installation steps 31
- Installation steps for PC-Client 48
- installer suddenly quits 102
- installing 37, 38
- interfaces file 67

K

- keywords 83
 - reserved by Sybase 61

L

- LDAP libraries
 - environment variables 51
 - location of 51

M

minimum size for database devices 113
 monitoring and diagnostic tables 41
 monitoring remote servers using MDA tables 41

N

new license server 13

O

objects
 conflicting names of 57
 renaming 57
 obtaining
 license 9
 ODBC 20
 OLE DB 20
 options for the server editions
 optional packages 2
 Overview of the upgrade process 53

P

PC_Client
 separate products 8
 PC-Client
 product description
 directory layout 7
 PCI, MDA tables 41
 planning the installation 9
 post downgrade 83
 prerequisites 3
 preupgrade 55
 privileges 113

Q

quoted identifier errors 107

R

raw partitions 113
 availability of 113
 re-running the upgrade 104
 recovering from a failed installation 102
 recovering from a failed upgrade 103

reenable Replication Server 67
 renaming objects 57
 replacing the binary 78
 replicated databases
 restoring 68
 replication issues with downgrade 85
 Replication Server 67
 draining logs for Adaptive Server 67
 reserved word conflicts 57
 reserved words 56
 checking for conflicts 58
 reserved words 58
 using quoted identifiers 58
 resource files 62
 response file 37, 49
 response file change 37
 restoring from backup 103

S

select* problem areas 108
 served license
 generating 11
 Server
 component descriptions 4
 directory layout 4
 server installation 3
 shared memory parameters
 adjusting 25
 silent mode 38
 source environment variables 31
 sp_checkreswords system procedure 57
 sqlupgrade utility 61
 sqlupgraderes utility 62
 stopping Adaptive Server after a failure 102
 stopping servers 102
 stored procedures
 changing after renaming objects 56
 style conventions
 fonts 3
 SYBASE directory 31
 sybsytemprocs database
 increasing the size of 125
 syntax examples 110
 syntax for using dbcc upgrade_object 109
 sysmessages 83
 system requirements
 Linux 17
 PC-Client 23
 System Requirements for PC-Client 48

system stored procedures 83
System tables 83

T

technical support
 calling 95
troubleshoot
 SySAM
 grace period 87
troubleshooting
 by using error log files 102

U

unattended mode 38
uninstalling 42
unknown cause of failure 103, 105
unrecognized tokens or objects 83

unserved license
 generating 12
upgrade high availability 40
upgrading compiled objects in database dumps
 111
upgrading compiled objects with dbcc
 upgrade_object 105
upgrading high availability and cluster support 76
upgrading Job Scheduler templates 72
Upgrading using BCP 78
utilities
 dsedit 67
 sqlupgrade 61
 sqlupgraderes 62

W

workflow 3

