Sun Java System Application Server Enterprise Edition 8.2 Standalone Installer Release Notes

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 819–4729–14
February 2009

090206@21990

◆ ◆ ◆ CHAPTER 1

Overview

The Sun Java™ System Application Server Enterprise Edition 8.2 product greatly simplifies the task of creating and administering Web services applications. It provides superior performance, clustering, and high availability features for scalable services that continue to operate despite software and hardware faults. The Application Server provides a development path for web services that simplifies the development process while providing uniquely flexible growth opportunities.

About These Notes

These Release Notes contain important information available at the time of release of Sun Java System Application Server 8.2 . Enhancements, known problems, and other late-breaking issues are addressed here. Read this document before you begin using Application Server Enterprise Edition8.2.

The most up-to-date version of these release notes can be found at the Sun Java System documentation Web site (`http://docs.sun.com/app/docs/coll/1310.4`). Check the Web site prior to installing and setting up your software and then periodically thereafter to view the most up-to-date release notes and product documentation.

Third-party URLs are referenced in this document and provide additional, related information.

Note – Sun is not responsible for the availability of third-party Web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Release Notes Revision History

This section lists the changes that have been made in these release notes after the initial release of the Sun Java SystemEnterprise Edition Application Server 8.2 product.

TABLE 1–1 Release Notes Revision History

Revision Date	Description
June 2005	Initial release of the Sun Java SystemApplication Server 8.2 product.
September 2005	Ongoing updates to known problems, fixed bugs, and various support URLs.
October 2005	Added information about Microsoft Windows support.
May 2008	Added “System Virtualization Support” section.

Accessibility Features

To obtain accessibility features that have been released since the publishing of this media, consult Section 508 product assessments available from Sun upon request to determine which versions are best suited for deploying accessible solutions. Updated versions of applications can be found at: `http://sun.com/software/javaenterprisesystem/get.html`

For information on Sun's commitment to accessibility, visit `http://sun.com/access`.

Related Documentation

The Sun Java System Application Server documentation set covers the following product releases:

- Application Server Enterprise Edition 8.2 2005Q1
- Application Server Enterprise Edition 8.2
- Application Server Enterprise Edition 8.2

Unless otherwise specified, whenever either product release is mentioned in the documentation, both releases are implied.

The following table summarizes the books included in the Application Server core application documentation set.

TABLE 1–2 Books in This Documentation Set

Book Title	Description
Documentation Center	**One stop location to access all Application Server topics.**
Quick Start Guide	How to get started with the Sun Java System Application Server product.
Installation Guide[1]	Installing the Sun Java System Application Server software and its components.
Deployment Planning Guide	Evaluating your system needs and enterprise to ensure that you deploy Sun Java System Application Server in a manner that best suits your site. General issues and concerns that you must be aware of when deploying an application server are also discussed.
Developer's Guide	Creating and implementing Java 2 Platform, Enterprise Edition (J2EE™ platform) applications intended to run on the Sun Java System Application Server that follow the open Java standards model for J2EE components and APIs. Includes general information about developer tools, security, assembly, deployment, debugging, and creating lifecycle modules.
J2EE 1.4 Tutorial	Using J2EE 1.4 platform technologies and APIs to develop J2EE applications and deploying the applications on the Sun Java System Application Server.
Administration Guide	Configuring, managing, and deploying the Sun Java System Application Server subsystems and components from the Administration Console.
High Availability Administration Guide	Post-installation configuration and administration instructions for the high-availability database.
Administration Reference	Editing the Sun Java System Application Server configuration file, `domain.xml`.
Upgrade and Migration Guide	Migrating your applications to the new Sun Java System Application Server programming model, specifically from Application Server 6.x and 7. This guide also describes differences between adjacent product releases and configuration options that can result in incompatibility with the product specifications.
Performance Tuning Guide	Tuning the Sun Java System Application Server to improve performance.
Troubleshooting Guide	Solving Sun Java System Application Server problems.
Error Message Reference	Solving Sun Java System Application Server error messages.

TABLE 1–2 Books in This Documentation Set *(Continued)*

Book Title	Description
Documentation Center	**One stop location to access all Application Server topics.**
Reference Manual	Utility commands available with the Sun Java System Application Server; written in manpage style. Includes the `asadmin` command line interface.

1 The Application Server *Installation Guide* is provided for a standalone installation of the Application Server.

How to Report Problems and Provide Feedback

If you have problems with Sun Java System Application Server, contact Sun customer support using one of the following mechanisms:

- Feedback Submittal form (`http://java.sun.com/docs/forms/J2EE14SubmittalForm.html`) — A form for submitting feedback on the Application Server product
- J2EE-INTEREST list (`http://archives.java.sun.com/archives/j2ee-interest.html`) — A mailing list for J2EE questions
- Bug database on Java Developer Connection (`http://developer.java.sun.com/servlet/SessionServlet?url=/developer/bugParade/index.jshtml`) — To view bugs or to submit a bug, use the Java Developer Connection Bug Parade
- Java Technology Forums (`http://forum.java.sun.com/`) — An interactive message board for sharing knowledge and questions about Java technologies and programming techniques; use the J2EE SDK forum for discussions related to the Sun Java System Application Server Enterprise Edition 8.2 product
- Sun Software Support services (`http://www.sun.com/service/sunone/software`) — Links to the Knowledge Base, Online Support Center, and Product Tracker, as well as to maintenance programs and support contact numbers
- The telephone dispatch number associated with your maintenance contract

  So that we can best assist you in resolving problems, please have the following information available when you contact support:
- Description of the problem, including the situation where the problem occurs and its impact on your operation
- Machine type, operating system version, and product version, including any patches and other software that might be affecting the problem
- Detailed steps on the methods you have used to reproduce the problem
- Any error logs or core dumps

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions.

To share your comments, go to `http://docs.sun.com` and click Send Comments. In the online form, provide the document title and part number. The part number is a seven-digit or nine-digit number that can be found on the title page of the book or at the top of the document. For example, the title of this book is *Sun Java System Application Server Enterprise Edition 8.2 Release Notes*, and the part number is 819-4729.

Additional Sun Resources

Useful information can be found at the following locations:

- Application Server product information (`http://wwws.sun.com/software/products/appsrvr/home_appsrvr.html`)
- Application Server product documentation (`http://docs.sun.com/app/docs/coll/1310.4`)
- Sun Java System Documentation (`http://docs.sun.com/prod/java.sys`)
- Sun Java System Professional Services (`http://www.sun.com/service/sunps/sunone`)
- Sun Java System Software Products and Service (`http://www.sun.com/software`)
- Sun Java System Software Support Services (`http://www.sun.com/service/sunone/software`)
- Sun Java System Support and Knowledge Base (`http://www.sun.com/service/support/software`)
- Sun Support and Training Services (`http://training.sun.com`)
- Sun Java System Consulting and Professional Services (`http://www.sun.com/service/sunps/sunone`)
- Sun Java System Developer Information (`http://developers.sun.com`)
- Sun Developer Support Services (`http://www.sun.com/developers/support`)
- Sun Java System Software Training (`http://www.sun.com/software/training`)
- Sun Software Data Sheets (`http://wwws.sun.com/software`)
- Sun Microsystems product documentation (`http://docs.sun.com/`)

◆ ◆ ◆ CHAPTER 2

About Application Server 8.2

The Sun Java System Application Server Enterprise Edition 8.2 is a J2EE 1.4 platform-compatible server for the development and deployment of J2EE applications and Java technology-based web services in large-scale production environments.

This section includes:

What's New in the 8.2 Release

The Application Server Enterprise Edition 8.2 includes the following enhancements:

- **Improved Administration** — The Application Server supports the remote secure management of complex multi-machine enterprise deployments using either a browser based console or a scriptable command line interface. It also provides a rich JMX based API allowing remote, secure, programmatic access to administrative and monitoring functions.
- **Message Broker** — The Application Server is bundled with an integrated enterprise class message broker that features providing highly available, reliable, high performance, and scalable messaging.
- **Expanded Platform Support** — Additional operating systems, databases, locales, and hardware are supported.

- **Sun Java Enterprise System** — As a key component of the Sun Java Enterprise System, the Application Server is tightly integrated with portal and network identity services.
- **Migration and Upgrade Tools** — These tools enable you to verify J2EE applications for standards conformance and portability, help with migrations from other J2EE Application Servers (JBoss, WebLogic, WebSphere), and aid in upgrading from previous versions of Sun ONE Application Server/ iPlanet Application Server.
- **Java 2 Standard Edition 5.0 Support** — The Application Server supports the Java 2 Standard Edition 5.0, which includes enhanced management and monitoring features and many performance and scalability improvements.
- **Java Web Services Developer Pack 1.6 (JWDSP) Plugin Support** — All JWSDP plugins are now supported. ..
- **JDBC Drivers** — The Application Server is bundled with Sun JDBC drivers.
- **Web Services Security** — These container message security mechanisms implement message-level authentication (for example, XML digital signature and encryption) of SOAP web services invocations using the X509 and username/password profiles of the OASIS WS-Security standard.
- **WS-I Basic Profile 1.1** — As mandated by the J2EE 1.4 specification, this release implements Web Services Interoperability (WS-I) Basic Profile 1.1 to enable interoperability for web services applications.
- **Backend Connectivity with iWay Adapters** — Sun Microsystems now resells and supports twenty-two iWay adapters to key backend systems (SAP, Siebel, Oracle, CICS, and IBM MQ Series) to help you leverage existing IT applications from within the Application Server environment. These adapters support the J2EE Connector Architecture 1.5 specification and Web services (SOAP) standards, and include developer tools to reduce time to connect to backend applications.
- **Latest HADB Management System** — The UNIX™ platforms contain the new high availability database (HADB) management system (HADB version 4.4.2–7), which includes a Database Server, ODBC 2.5 driver, JDBC 3.0 type 4 driver, `clusql` (an interactive program to enter and execute SQL statements), and a management system. This version eliminates the dependency on SSH/RSH, but requires that the network be configured for UDP multicast. See the *Sun Java System Application Server Enterprise Edition 8.2 High Availability Administration Guide* for the details on HADB requirements and limitations.
- **Solaris 10 Zones Support** – The Application Server can be installed in either a global or non-global Zone on Solaris 10 systems. See the Solaris Zones (`http://www.sun.com/bigadmin/content/zones/`) page for more information about Solaris Zones.

Hardware and Software Requirements

This section lists the requirements that must be met before installing the Sun Java System Application Server Enterprise Edition 8.2 product.

Platform Requirements

The following table lists the operating systems that are supported for Sun Java System Application Server Enterprise Edition 8.2 product. Additionally, the minimum and recommended memory requirements are identified for installing and running the Application Server.

TABLE 2–1 Sun Java System Application Server 8.2 Platform Requirements

Operating System	Minimum Memory	Recommended Memory	Minimum Disk Space	Recommended Disk Space	JVM
Sun Solaris 8, 9, 10 (SPARC) Solaris 9, 10 (x86)	512 MB	1 GB	250 MB free	500 MB free	J2SE 1.4.2_06, J2SE 5.0
Red Hat Enterprise Linux 2.1 Update 2, 3.0 Update 1	512 MB	1 GB	220 MB free	300 MB free	J2SE 1.4.2_06, J2SE 5.0
Windows Server 2000 SP4+ Windows 2000 Advanced Server SP4+ Windows Server 2003 Windows XP Pro SP1+	1 GB	2 GB	500 MB free	1 GB free	J2SE 1.4.2_06, J2SE 5.0

On UNIX, you can check your operating system version using the `uname` command. Disk space can be checked using the `df` command.

System Virtualization Support

System virtualization is a technology that enables multiple operating system (OS) instances to execute independently on shared hardware. Functionally, software deployed to an OS hosted in a virtualized environment is generally unaware that the underlying platform has been virtualized. Sun performs testing of its Sun Java System products on select system virtualization and OS combinations to help validate that the Sun Java System products continue to function on properly sized and configured virtualized environments as they do on non-virtualized systems. For information about Sun support for Sun Java System products in virtualized environments, see System Virtualization Support in Sun Java System Products.

Important Patch Information

For the current list of required patches for Sun Java System Application Server Enterprise Edition 8.2 go to `http://sunsolve.sun.com` and search for "`app server 8.1 patch`." Follow the Sun Java System Application Server Enterprise Edition 8.2 links. As operating system patch requirements change and patches to Java Enterprise System components become available, updates will be made available on SunSolve, initially in the form of recommended patch clusters.

Solaris Patch Requirements

It is recommended that Solaris 9, 10 (x86, SPARC) users have the "Sun recommended patch cluster" installed. This patch cluster is available under Recommended and Security Patches (`http://sunsolve.sun.com/`) on SunSolve.

RedHat Enterprise Linux 3.0 Additional Package Requirements

To run native components of this product, including installer, the following package, which is not part of the standard RedHat Enterprise Linux 3.0 distribution, should be installed:
`compat-libstdc++-7.3-2.96.118.i386.rpm`

The package can be downloaded from the Redhat Linux web site.

JDBC Drivers and Databases

The Sun Java System Application Server is designed to support connectivity to any DBMS with a corresponding JDBC driver. For a list of components that Sun has tested and found to be acceptable for constructing J2EE compatible database configurations, please refer to the following table:

TABLE 2–2 J2EE-Compatible JDBC Drivers

JDBC Vendor	JDBC Driver Type	Supported Database Server
i-net Software	Type 4	Oracle (R) 8.1.7, 9i, 9.2.0.3 Sybase ASE 12.5.2 Microsoft SQL Server 2000 4.0 Service Pack 1
IBM	Type 2	IBM DB2 8.1 Service Pack 3+
PointBase	Type 4	PointBase Network Server 4.8
DataDirect	Type 4	Oracle (R) 8.1.7, 9i, 9.2.0.3 Sybase ASE 12.5.2 Microsoft SQL Server IBM DB2 8.1 Service Pack 3+
Sun Java System JDBC Driver for Oracle	Type 4	Oracle (R) 9.2.0.3, 10G
Sun Java System JDBC Driver for DB2	Type 4	IBM DB2 8.1 Service Pack 3+
Sun Java System JDBC Driver for Sybase	Type 4	Sybase ASE 12.5.2
Sun Java System JDBC Driver for Microsoft SQL Server	Type 4	Microsoft SQL Server 2000 4.0 Service Pack 1
Oracle	Type 4, Type 2	Oracle (R) 9.2.0.3, 10G

For more information about i-net Software, see `http://www.inetsoftware.de/`.

For more information about DataDirect Technologies, see `http://www.datadirect.com/`.

Configuring Oracle

Oracle JDBC drivers must be configured properly to be compliant with J2EE 1.4. Use the following configuration for Type 2 and Type 4 drivers:

▼ To configure Oracle

1 **Use the JDBC driver from 9.2.0.3 or later.**

2 **The Oracle database needs to have `compatible=9.0.0.0.0` or higher in its parameter (`init.ora`) file.**

3. **Use the `ojdbc14.jar` file.**

4. **Configure the Application Server to define the following JVM property:**

   ```
   -Doracle.jdbc.J2EE13Compliant=true
   ```

   In addition, for `Type-2` drivers, both the `ORACLE_HOME` and `LD_LIBRARY_PATH` variables (which must include `$ORACLE_HOME/lib`) need to be defined in the environment in which the Application Server is started. For example, add them to the `asenv.conf` file and ensure they are exported.

Configuring PointBase

Many sample applications use the PointBase database server included with the Application Server. When using Application Server Enterprise Edition, you must configure the PointBase database server before using it.

There are two ways to configure PointBase:

1. Using the command appropriate for your operating system and shell, set the `JAVA_HOME` environment variable to the directory where J2SE is installed. For example: `% setenv JAVA_HOME "/opt/SUNWappserver/jdk"`
2. Edit the Application Server's PointBase configuration file as follows:
   a. On Solaris and Linux systems, edit the *install_dir*`/pointbase/tools/serveroption/pbenv.conf` configuration file, changing the line:

      `PB_JAVA=%%%PB_JAVA%%%`

      to

      `PB_JAVA=`*J2SE_location*

   a. On Windows systems, edit the *install_dir*`\pointbase\tools\serveroption\pbenv.bat` configuration changing the line:

      `PB_JAVA=%%%PB_JAVA%%%`

      to

      `PB_JAVA=`*J2SE_location*

   Where *J2SE_location* is the directory where the J2SE is installed. If you installed J2SE with Application Server, it is installed by default to *install_dir*`/jdk`.

   After making this change, you can start PointBase using the `startserver` script.

Web Servers

This section lists the web servers that are supported for the Sun Java System Application Server Enterprise Edition 8.2 .

TABLE 2–3 Supported Web Servers

Web Server	Version	Operating System
Sun Java System Web Server	6.1+	Solaris SPARC 8, 9, 10 Solaris *x*86 9, 10 Red Hat Enterprise Linux 2.1 Update 2, 3.0 Update 1
Apache Web Server	1.3+, 1.4, 2.0	Solaris SPARC 9, 10 Solaris *x*86 10 Red Hat Enterprise Linux 2.1 Update 2, 3.0 Update 1 Windows Server 2003 Windows 2000 Advanced Server SP4+ Windows Server 2000 SP4+ Windows XP Pro SP1+
Microsoft IIS®	5.0+	Windows Server 2003 Windows 2000 Advanced Server SP4+ Windows Server 2000 SP4+ Windows XP Pro SP1+

Browsers

This section lists the browsers that are supported with the Sun Java System Application Server Enterprise Edition 8.2 .

TABLE 2–4 Supported Web Browsers

Browser	Version
Mozilla	1.4, 1.5, 1.6, 1.7.*x*
Netscape Navigator	4.79, 6.2, 7.0
Internet Explorer	5.5 Service Pack 2, 6.0

HADB Requirements and Supported Platforms

In addition to the requirements listed in "Hardware and Software Requirements" on page 11, verify that your system meets the requirements listed below for running HADB.

Note – The Java components of the system have been built with JDK 1.4.2_02 and has been tested on JDK 1.5.

Supported Platforms

- **Solaris (SPARC)** – Solaris 8 MU7, Solaris 9 MU7, Solaris 10 RR.
- **Solaris (x86)** – Solaris 9 MU7, Solaris 10 RR.
- **RedHat Enterprise Linux** - 2.1 U5 (only `ext2` file system is supported, not `ext3`), 3.0 U4 (both `ext2` and `ext3` are supported. Updates before U4 are not recommended due to excessive swapping). Note that HADB is tested on these operating system versions in 32–bit mode only. Also, note that HADB does not support RedHat Enterprise Linux 3.0 running in 64–bit mode due to a bug in the operating system (see known bug 6249685 in the "High Availability" on page 48 section for details about impact on HADB).
- **Microsoft Windows** – Microsoft Windows 2000 Advanced Server Service Pack 4 and Microsoft Windows 2003 Enterprise Edition. Note that HADB does not support any of the forthcoming Microsoft Windows operating system versions in 64–bit mode.

HADB Server Host Requirements

- **Minimum memory** - 320 MB per node.
- **Minimum free disk space** - 70 MB for HADB binaries per host. In addition, disk space is needed for the data devices, 512 MB for a test installation per node.
- **Recommended memory** - 512 MB per node.
- **Recommended free disk space** - 70 MB for HADB binaries per host. In addition, disk space is needed for the data devices, 1200 MB for a test installation per node.

Note – Make sure write caching is disabled on devices storing HADB data and log files. Write caching is enabled by default on some Solaris platforms; for example, Solaris *x*86.

HADB Management Host Requirements

- **Minimum memory** - 128 MB
- **Minimum free disk space** - 70 MB for HADB binaries per node

HADB Client Host Requirements

- **Minimum memory** - 120 MB
- **Minimum free disk space** - 20 MB

Upgrading the Sun Java System Application Server

The in-place upgrade from any prior Application Server release is not supported. Refer to the *Sun Java System Application Server Enterprise Edition 8.2 Upgrade and Migration Guide*for complete instructions for upgrading from a previous version of the Application Server to the current version.

Switching to a Supported J2SE Version

If you want to use PointBase with the Application Server, download J2SE 1.4.2 and use it instead of the bundled J2SE 5.0 JVM. To do this perform the following steps:

▼ To switch to J2SE 1.4.2

1. **Download the J2SE 5.0 SDK (not the JRE) from and install it on your system, if you have not already done so.**

   The J2SE 5.0 SDK can be downloaded from `http://java.sun.com/j2se/1.5.0/`

2. **Stop the Application Server.**

   From the command line:

   install_dir`/bin/asadmin stop-domain`

   From the Administration Console:

   a. **Click the Application Server node.**

   b. **Click Stop Instance.**

3. **Edit the** *install_dir*`/config/asenv.conf` **file (**`asenv.bat` **on Windows), changing the value for** `AS_JAVA` **to point to the J2SE 1.4.2 home directory:**

4. **Edit the** *as-install*`/samples/common.properties` **file, changing the line beginning** `com.sun.aas.javaRoot...` **to reference the J2SE 1.4.2 home directory.**

5 **Restart the Application Server.**

From the command line:

install_dir`/bin/asadmin start-domain`

From the Administration Console:

a. **Click the Application Server node.**

b. **Click Start Instance.**

Other Requirements

The following additional requirements should be met before installing the Sun Java System Application Server software.

- **Free space:** your temporary directory must have a minimum of 35MB free for Sun Java System Application Server installation, and 250 MB of free space for the SDK installation.
- **Using the uninstall program:** If you need to remove the Application Server from your system, it is important to use the uninstall program that is included with the software. If you attempt to use another method, problems will arise when you try to reinstall the same version, or when you install a new version.
- **Free ports:** You must have seven unused ports available.
  - The installation program automatically detects ports in use and suggests currently unused ports for the default settings. By default, the initial default ports are `8080` for HTTP, `8181` for HTTPS, and `4849` for the Administration Server.
  - The installation program will detect used ports and assign two others for you: Sun Java System Message Queue (by default, 7676), and IIOP (by default, 3700 for IIOP and 1060 and 1061 for IIOP/SSL). If these default port numbers are in use, the installation program will assign a random port number from the dynamic port range (note that this may not be the next available port number).

  Starting previously-installed servers (UNIX) — unless you are replacing the previously installed server, you should start it before you begin the Sun Java System Application Server 8.2 installation process. This allows the installation program to detect ports that are in use and avoid assigning them for other uses.
- **Replacing previously-installed servers** (UNIX) — if you have an older version on the Sun Java System Application Server installed that you wish to replace with the current Application Server, you should stop it before installing the new server. Use the installation program upgrade wizard to upgrade the server.
- **Shutting down firewall** (Microsoft Windows) — You must stop any firewall software before installing the Sun Java System Application Server software, because some of this software disables all ports by default. The installation program must be able to accurately determine which ports are available.

For further compatibility information, see the *Sun Java System Application Server Enterprise Edition 8.2 Upgrade and Migration Guide*.

Bugs Fixed in the Enterprise Edition 8.2 Release

This section lists the customer-escalated issues resolved for the Sun Java System Application Server Enterprise Edition 8.2 product.

Bug Number	Description
4887079	Programmatic APIs for deploying/undeploying and finding out which applications are deployed.
4911462	Incorrect message when port outside of available range.
4918535	`sun-appserv-deploy()` doesn't take flag to support `createAndDropTables()`
4939749	`xml:lang()` value should not be inserted automatically by the Deploytool.
4946914	Deployment support for cluster.
4979136	Directory based deployment copies the application in a backup directory.
4987274	Deployment fails if remote interface for the bean is named `Util()`
4988818	Transparent Persistence runtime tests fail when J2SE 1.5 is used.
4992295	Deploying a system component succeeds on the command-line interface, but an error is logged in server's log file.
4994790	JSP deployed with *precompilejsp=true* does not use compiler flags in `sun-web.xml`.
4996876	Verifier versus deploy with *verify=true*, different reports.
5003356	Recent `server.policy` file updates are not accounted by Upgradetool.
5006854	`asadmin deploy --virtualservers` fails to deploy.
5007309	Inappropriate default value for HTTP listener acceptor threads.
5008941	JSR88 start operation fails when an application is deployed again after it is undeployed.
5016848	On Windows, the JDK JAR file caching and unclosed files prevent some redeployments.
5017956	`list -m` at the JAR module level does not list the EJB's.
5030425	The `deploydir` command ignores `security-role-mapping` changes.
5041343	Not checking `servlet-mapping url-pattern--directory` always wrapped by /.

Bug Number	Description
5046120	SEVERE log messages while deploying big applications.
6041268	No mechanism to disable `HTTP TRACE`.
6062410	Upgradetool is launched in English on a localized machine.
6067341	The `deploydir` command on a web application with `ejb-refs` fails on `rmic` the remote interfaces.
6152752	An `outofbound` exception is logged during SPEC J2004 test runs.
6154949	Connection Validation does not work.
6157310	Runtime reloads the `Collection` field during relationship management.
6165491	Failed to start a domain if it was created on a different path than the default domain.
6171667	The lifecycle modules property elements are not created in `domain.xml`.
6171729	Non-String `RA ActivationSpec` properties result in a `IllegalArgumentException` during MDB deployment.
6172178	OSS/J TT TCK failed to get JMS connection factory from a remote application server.
6172589	Optimize calls to security manager.
6183492	[DataDirect] DB2: Some trasparent persistence application server tests failed with exception thrown during EJB invocation.
6184864	EJB QL query returns no results using OR operator and expression contains null single-valued CMRS.
6197393	The Deploytool often won't create message-destination element in deployment descriptor.
6198796	EE samples `asadmin` commands needs to include `availabilityenabled=true()` option when deploying the application.
6198981	Missing `xalan.jar` file from classpath causes empty drop-downs and web service wizard to fail.
6199076	Unable to run the Duke bookstore sample failover test with the `asant` script.
6202363	Cluster name hard coded in one `ant` target in `mq-failover` sample application.
6202606	JMS service configuration cannot be used for SSL JMS between JMS and Message Queue.
6206176	Application Server 8.1 requires `startserv/stopserv` to have permissions of 755.
6207297	Accessing the Application Server without the default SSL Port number (443) does not work.

Bug Number	Description
6207862	`asadmin create-domain --help` produces some is garbled.

Bugs Fixed in the Enterprise Edition 8.2 Release

This section lists the customer-escalated issues resolved for the Sun Java System Application ServerEnterprise Edition8.2product.

Bug Number	Description
4842830	The exception “ComStream is closed” reaches JDBC client.
4847716	One should not use execute/executeUpdate to set commit mode as this may cause unwanted behavior. Use the standard JDBC `setAutocommit()`.
4861326	Statement pool does not recognize CREATE SCHEMA as an implicit SET SCHEMA.
4891060	Listeners ignore address directive when listening on sockets.
5042351	New tables created after new nodes are added will not be distributed on the added nodes.
5061316	Queries against a table being refragmented may fail with HADB-E-01792: Replica has been deleted. The query must be retried.
5063175	`hadbm create` should give error when using host with both single/multiple nets.
5079029	Unregistering a package on a single host may fail with the error “The software package is in use by a database instance and can not be removed”
5094611	Management operations which require opening a write transaction in the administration repository, may in very rare cases, hang forever waiting for the transaction to open.
5103186	NSUP unable to start with one net down on Windows 2003.
6225613	
6271063	Install/removal of HADB c package (Solaris: SUNWhadbc, Linux: sun-hadb-c) version symlink `/opt/SUNWhadb/` causes error.
6174781	The command `hadbm` status – nodes may show that nodes have a `nodestate` equal to Unknown in a short period after the management agents have been restarted.
6175436	If `hadbm addnodes` or `hadbm refragment` fails with error HADB-E-11747: Nodegroup all_nodes exists already, run `hadbm refragment` again.

Bug Number	Description
6174676 6179084	Unable to run configure-ha-cluster.
6178228 6179010	Failure in configure-ha-cluster
6181845	Cannot create one data device larger than 2 GB on Windows.
6189189	`export-http-lb-config` doesn't create `loadbalancer.xml` file name when absolute path is given.
6198225	QuickStart Guide has typo; i.e., repeated sentence.
6195779	The values of options of some filter drop-down lists are not internationalized.
6196741	Upgrade in place of bundled J2SE component does not work correctly when upgrading J2SE 1.4.x.
6207616	If a host is down, any `hadbm` command may hang for minutes if it needs to connect to the management agent.
6212791	No items displayed in the right pane when clicking a tree node.
6216096	A hanging transaction may cause node crash due to “log buffer full” and a large number of transactions being aborted.
6225613	Inconsistent LOB size in executeUpdate()
6227502	Initialization errors in EJB Timer Service should not be logged as `SEVERE`.
6228789	`hadbm delete` command fails.
6230415	HADB-E-21070: The operation did not complete within the time limit, but has not been cancelled and may complete at a later time.
6230792	hadbm:Error 22009: The command issued had no progress in the last 300 seconds.
6232347	`dropandcreatetables` not quite correct for `asdamin deploy --help`.
6232838	Unneeded log calls preventing application server from scaling.
6232974	Installer failed to create a node agent when upgrading from 8.0 Platform Edition to 8.1 Enterprise Edition.
6233142	Installing/uninstalling HADB should always preserve the softlink /opt/SUNWhadb/4, but, this has not always been the case.
6233276	Form authorization does not work for URL `-pattern /*.jsp`.
6233469	Incorrect help text in `asadmin` help.
6233476	Incorrect help text for `update-file-user` and similar commands.

Bug Number	Description
6237567	Missing key `adminObjectStep2PageHelp` in Create Administration Object Resource window.
6238477	Unable to resolve "corba name" EJB references in same application server instance.
6239630	Cannot properly map a particular entity bean.
6239837	Wrong unit and default value for Reconnect Interval in Administration for JMS `server-config`.
6240661	some messages remain in english in the locale.
6241311	The note for "Pool Idle Timeout" field is wrong.
6241368	Admin Console login screen and online help never refer to English browser language.
6243395	Transaction recovery not working with a JMS and JDBC resource.
6245922	Application Server crashes consistently.
6246426	Expansion of JAR files in WEB-INF/lib exposes content that should be concealed.
6249637	Change of JDBC connection pool properties requires a restart.
6249662	`Proxy-auth-cert` not properly formatted.
6250989	SOAP Element.`addChildElement` adds an incomplete element without markups.
6252187	High Availability single sign-on propagates principals between different realms.
6252810	`configure-ha-persistence` in man pages is not up-to-date.
6253735	QuickStart does not include High Availability information.
6254393	Bundled QuickStart has link to out-of-date release notes.
6254462	NPE thrown by connection validation code after database restart.
6255253	"How to Buy" link in bundled documentation is wrong URL.
6255440	Performance improvement for synchronization.
6255458	Typo in `delete-virtual-server`.
6255524	`UpdateTask` ANT task does not work with ANT 1.6.2.
6255564	Upgraded domain startup fails due to administration user authentication failure after Platform Edition to Enterprise Edition upgrade.
6258844	File realm user login does not work after upgrade to 8.1 Update 1.
6258997	Correct description for `--secure` option in command—line interface man pages.

Bug Number	Description
6259125	Documentation for `asadmin get` is inadequate and confusing.
6262564	`PrivateKeyProcessor` does not support get by `keyIdentifier`.
6262824	Solaris 10: Stopping a management agent by using the ma-initd script in a global zone stops the management agent in the local zone as well.
6263684	Linux RPM patch generation script requires manual changes in the `README`.
6263686	svr4 patch generation inserts incorrect entries in `README`.
	`package-appclient` script broken.
6264969	The setup of all AS samples that use PointBase failed: Cannot upgrade database to version 5.1.
6265687	Installer's graphic image has incorrect product version.
6266183	High Availability test failure: Realm name after restart contains null.
6267410	Exception during `session.invalidate()` if Log level is set to FINE.

Additional HADB Information

This section describes important additional information about the HADB implementation included in Application Server 8.2.

HADB Enhancements

- A new management command `hadbm setadminpassword` has been implemented to allow changing the password used for database administration. The command takes options indicating which management agent to use, and the old and new password. For more information, see the `hadbm setadminpassword` man page.
- The existing management command `hadbm listpackages` has been modified. Previously, the command took no operands, and listed all packages in the relevant management domain. The modifications introduces an optional package name operand, and lists only packages with that name. If the operand is not provided, all packages are listed. For more information, see the `hadbm listpackages` manpage.

- The existing management command `hadbm createdomain` has been modified. The *hostlist* operand is extended to also specify the port number of the management agent. In this way, the domain is completely specified using only the *hostlist* operand. The old behavior is still supported for backward compatibility. For more information, see the `hadbm createdomain` manpage.
- Some of the error messages from the management system have been modified. The modifications are intended to improve understandability, consistency and accuracy of the error messages. The actual modifications are not listed in these release notes.
- The installation and uninstallation behavior has been slightly changed. Installing or uninstalling the HADB should always preserve the softlink `/opt/SUNWhadb/4`, but this has not always been the case:
- The possibility of entering passwords on the command line as a command option is deprecated. This is relevant to all `hadbm` commands taking passwords as command line options. For `hadbm` commands, it has previously been possible to enter a password as:

  1. A password file
  2. A command line option
  3. An interactive input

  Method 2, the command line option, is considered unsafe, and is therefore deprecated. A warning message is issued if a password is entered in this way. Instead, use a method 1, password file, or method 3, interactive output. Using a password at the command line will become obsolete in the next release. Note this applies to all `hadbm` commands taking a command line password option.
- HADB has been upgraded to use JGroups Version 2.2, and its source code is distributed along with the HADB. To support online upgrade from a previous HADB version, both JGroups 2.1 and 2.2 are delivered with HADB. For JGroups 2.1, byte code is delivered only.

HADB File System Support

There are several important considerations if you want to configure HADB to use one of the following file systems:

- **ext2 and ext3**– HADB supports ext2 and ext3 file systems for Red Hat Application Server 3.0. For Red Hat Application Server 2.1, HADB supports only the ext2 file system.
- `Veritas`– When the Veritas File System is used on the Solaris platform, the message "`WRN: Direct disk I/O mapping failed`" is written to the history files. This message indicates that HADB cannot turn on direct I/O for the data and log devices. Direct I/O is a performance enhancement that reduces the CPU cost of writing disk pages. It also causes less overhead of administering dirty data pages in the operating system.

  To use direct I/O with the Veritas File System, use one of the following:

- Create the data and log devices on a file system that is mounted with the option `mincache=direct`. This option applies to all files created on the file system. See the `mount_vxfs(1M)` command for details.
- Use the Veritas Quick I/O facility to perform raw I/O to file system files. See the *VERITAS File System 4.0 Administrator's Guide for Solaris* for details.

  Note that these configurations have not been tested with Application Server 8.2 .

  Refer to the *Application Server Enterprise Edition High Availability Administration Guide* for information about installing and configuring HADB with Application Server software.

Upgrading the High Availability Database

Pre-upgrade Tasks/Data Migration

Before You Begin Users should keep the HADB history files, management agent configuration files, log files and repository, and all the data devices outside the installation path. If not, this should be done prior to the upgrade. To move the management repository and configuration files:

1. **Stop all the old management agents and keep the HADB nodes running.**
2. **On each host, move the repository directory to the new location.**
3. **On each host, copy the `dbconfig` directory to the new location.**
4. **On each host, update the `mgt.cfg` file, and set the correct path for `dbconfig` and repository directory.**
5. **Start the management agents using the updated `mgt.cfg` file.**

Upgrade Procedure

To upgrade from HADB version 4.4.x to version 4.4.2-7, apply the following steps:

1. **Perform the pre-upgrade tasks mentioned above as necessary.**
2. **Install HADB version 4.4.2-7 on all HADB hosts (on another path than that of version 4.4.x, for instance on `/opt/SUNWhadb/4.4.2-7`).**

3. **Install the HADB 4.4.2-7 version on the `hadbm` client hosts, if they are different than that of the HADB hosts.**

4. **Stop all management agents running on all HADB hosts.**

5. **Start the management agent processes using the new version's software, but with the old configuration files. In the remaining steps, please use the `hadbm` command found in the new version's `bin` directory.**

6. **Register the package in the management domain (default package name becomes V4.4, so another package name may be required to avoid conflicts with existing packages having the same name):**

   ```
   hadbm registerpackage --packagepath=/opt/SUNWhadb/4.4.2-7 V4.4.2-7
   ```

7. **Run the `hadbm listpackages` command and check that the new package is registered in the domain.**

8. **Restart the database with the new `hadbm` version 4.4.2-7. If it is necessary to move the devices and history files, run online upgrade combined with setting new paths for devices and history files in one single operation:**

   ```
   hadbm set packagename=V4.4.2-7,devicepath=new_devpath,
   historypath=new_histpath
   ```

   Otherwise, if the devices and history files are already outside of the installation directory, run the following command, which only does a rolling restart of the nodes:

   ```
   hadbm set packagename=V4.4.2-7 database name
   ```

9. **Check that the database status is "running" (using the `hadbm status` command) and that it functions normally, serving the client transactions.**

10. **If everything is working, the old installation can be removed later. Before unregistering the old package, remove all references to the old package from the `ma` repository. Otherwise, `hadbm unregisterpackage` will fail with "package in use." A dummy reconfiguration operation, for instance, `hadbm set connectiontrace=`*same as previous value* will remove all references to the old package. Now, unregister the old package:**

    ```
    hadbm unregisterpackage [--hosts=host-list] old pacakge name
    ```

11. **Remove the old installation from the file system.**

▼ Testing the Upgrade

On Solaris, to test that the upgrade was successful, check that the upgrade was performed properly:

1. **Ensure that the running processes use the new binaries. Check the following in all HADB nodes:**

   new path`/bin/ma -v`
   new path`/bin/hadbm -v`

2. **Check whether the database is running. The following command should show that all the HADB nodes are in a "running" state.**

   new path`/bin/hadbm status -n`

3. **Ensure that the products using HADB have changed their pointers to point to the new HADB path.**

4. **The products using the HADB can run their upgrade tests to verify the HADB upgrade is also working.**

   After an online upgrade, if the new version does not work properly, go back to using the previous HADB version. However, if there has been a change to the management agent repository, the HADB itself can be downgraded, but the new management agent must be kept running.

Special Deployment and Upgrade Information

This section lists additional information about HADB deployment and upgrading.

- "Deployment" on page 28
- "Online Upgrade from 4.4.1 to 4.4.2" on page 31

Deployment

- Store device, log and history files on local disks only, do not use remote-mounted file systems.
- If more than one node is placed on a host, it is recommended to keep the devices belonging to each node on different disks. Otherwise, the disk contention would reduce the performance. Symptoms of this problem can be seen in the history files by the messages like this: "BEWARE - last flush/fputs took too long." When one single node has more than one data device file, it is recommended to use separate disks for these device files.
- Use local disks (preferably separate disk than the one used for data devices) to install HADB binaries on HADB hosts. NFS delays or disk contention may cause node restarts with warning, "Process blocked for nnn, max block time is nnn" in the history files.
- Do not place the hadb devices, history files, management agent directories and agent config files in the hadb package path. This will cause problems when upgrading to newer versions and deleting the old package path.
- This release of HADB is offically supported for a maximum of 28 nodes; 24 active data nodes with 4 spares.
- We recommend using the same version for the JDBC driver and the HADB server.

- We do not support IPv6, only IPv4.
- The command line length on Windows is restricted to 2048 bytes.
- The network must be configured for UDP multicast.
- Due to excessive swapping observed in RedHat Enterprise Linux 3.0, updates 1 through 3, we do not recommend it as a deployment platform. The problem is fixed in RedHat Enterprise Linux 3.0 update 4.
- Possibility of running `NSUP` with real time priority:

  The node supervisor (`NSUP`) processes (`clu_nsup_srv`) ensure the high availability of the HADB with the help of exchanging "heartbeat" messages in a timely manner. The timing gets affected when an `NSUP` is colocated with other processes causing resource starvation. The consequence is false network partitioning and node restarts (preceded by a warning "Process blocked for `n` seconds" in history files) resulting in aborted transactions and other exceptions.

  To solve this problem, `clu_nsup_srv` (found in `installpath/lib/server`) must have the `suid` bit set and the file must be owned by root. This is achieved manually by the commands:

  ```
  # chown root clu_nsup_srv
  # chmod u+s clu_nsup_srv
  ```

  This causes the `clu_nsup_srv` process to run as the user `root` when started, and this in turn allows the process to automatically give itself real-time priority after startup. To avoid any security impact by using `setuid`, the real-time priority is set in the very beginning and the process falls back to the effective uid once the priority has been changed. Other HADB processes will lower their priority to timeshare priority.

  If NSUP could not set the real-time priority, it issues a warning, "Could not set realtime priority" (`unix: errno will be set to EPERM`), which is written out in `ma.log` file and continues without real-time priority.

  There are cases where it is not possible to set real-time priorities; for example:

  - When installed in Solaris 10 non-global zones
  - When `PRIV_PROC_LOCK_MEMORY` (Allow a process to lock pages in physical memory) and/or `PRIV_PROC_PRIOCNTL` privileges are revoked in Solaris 10
  - Users turn off `setuid` permission
  - Users install the software as tar files (nonroot install option for the `App.server`)

  The `clu_nsup_srv` process is not CPU consuming, its footprint is small and running it with real-time priority will not impact performance.
- Configuring IP network multipathing for HADB for Solaris (tested on Solaris 9 only):

  Sun recommends that Solaris hosts running HADB be set up with network multipathing in order to ensure the highest possible network availability. Network multipathing setup is covered in detail in the *IP Network Multipathing Administration Guide*. If you decide to use

multipathing with HADB, refer to the Administering Network Multipathing section of the *IP Network Multipathing Administration Guide* in order to set up multipathing before you proceed with adapting the multipathing setup for HADB as described below. The *IP Network Multipathing Administration Guide* is part of the Solaris 9 System Administrator Collection, and can be downloaded from `http://docs.sun.com`.

- **Set network interface failure detection time**

  For HADB to properly support multipathing failover, the network interface failure detection time must not exceed 1000 milliseconds as specified by the `FAILURE_DETECTION_TIME` parameter in `/etc/default/mpathd`. Edit the file and change the value of this parameter to `1000` if the original value is higher:

  ```
  FAILURE_DETECTION_TIME=1000
  ```

  In order for the change to take effect, issue the following command:

  ```
  pkill -HUP in.mpathd
  ```

- **IP addresses to use with HADB**

  As described in the *Solaris IP Network Multipathing Administration Guide*, multipathing involves grouping physical network interfaces into multipath interface groups. Each physical interface in such a group has two IP addresses associated with it: a physical interface address and a test address. Only the physical interface address can be used for transmitting data, while the test address is for Solaris internal use only. When `hadbm create --hosts` is run, each host should be specified with only one physical interface address from the multipath group.

- **Example**

  Assume that Host 1 and Host 2 have two physical network interfaces each. On each host, these two interfaces are set up as a multipath group, and running `ifconfig -a` yields the following:

  Host 1:

  ```
  bge0: flags=1000843<mtu 1500 index 5 inet 129.159.115.10 netmask ffffff00
  broadcast 129.159.115.255 groupname mp0
  bge0:1: flags=9040843<mtu 1500 index 5 inet 129.159.115.11 netmask ffffff00
  broadcast 129.159.115.255
  bge1: flags=1000843<mtu 1500 index 6 inet 129.159.115.12 netmask ffffff00
  broadcast 129.159.115.255 groupname mp0
  bge1:1: flags=9040843<mtu 1500 index 6 inet 129.159.115.13 netmask ff000000
  broadcast 129.159.115.255
  ```

  Host 2:

  ```
  bge0: flags=1000843<mtu 1500 index 3 inet 129.159.115.20 netmask ffffff00
  broadcast 129.159.115.255 groupname mp0
  bge0:1: flags=9040843<mtu 1500 index 3 inet 129.159.115.21 netmask ff000000
  ```

```
broadcast 129.159.115.255
bge1: flags=1000843<mtu 1500 index 4 inet 129.159.115.22 netmask ffffff00
broadcast 129.159.115.255 groupname mp0
bge1:1: flags=9040843<mtu 1500 index 4 inet 129.159.115.23 netmask ff000000
broadcast 129.159.115.255
```

Here, the physical network interfaces on both hosts are the ones listed as `bge0` and `bge1`. The ones listed as `bge0:1` and `bge1:1` are multipath test interfaces (they are thus marked as DEPRECATED in the `ifconfig` output), as described in the *IP Network Multipathing Administration Guide*.

To set up HADB in this environment, select one physical interface address from each host. In this example. we choose `129.159.115.10` from host 1 and `129.159.115.20` from host 2. To create a database with one database node per host, use the following argument to `hadbm create`:

```
--host 129.159.115.10,129.159.115.20
```

To create a database with two database nodes on each host, use the following argument:

```
--host 129.159.115.10,129.159.115.20,129.159.115.10,129.159.115.20
```

In both cases, the `ma.server.mainternal.interfaces` variable on both hosts should be set to `129.159.115.0/24`.

Online Upgrade from 4.4.1 to 4.4.2

It is not possible to upgrade from 4.2 or 4.3 to 4.4 online. However, 4.4 supports online upgrade for the future versions. To upgrade from 4.4.1 to 4.4.2, apply the following steps:

1. Install 4.4.2 on all HADB hosts (On another path than that of 4.4.1 – for instance `/opt/SUNWhadb/4.4.2-6`).
2. Install the new version on the `hadbm client` hosts.
3. Stop all management agents running on the HADB hosts.
4. Start the management agent processes using the new version's software, but with the old configuration files. In the remaining steps, please use the `hadbm` command found in the new version's `bin` directory.
5. Register the package in the management domain (default package name here becomes `V4.4`, so another package name may be required to avoid conflicts with existing packages having the same name):

   ```
   hadbm registerpackage --packagepath=/opt/SUNWhadb/4.4.2-6 V4.4.2
   ```

6. Restart the database with the new version (the following command does a rolling restart of the nodes):

   ```
   hadbm set packagename=V4.4.2 database_name
   ```

7. Check that the database status is "running" (using the command `hadbm status`) and that it functions normally, serving the client transactions.
8. If everything works, the old installation can be removed later:

   Before unregistering the old package, remove all references to the old package from the `ma` repository. Otherwise, `hadbm unregisterpackage` will fail with "package in use." A dummy reconfiguration operation, for instance, hadbm set connectiontrace=<same_as_previous_value> will remove all references to the old package. Now, unregister the old package:

   ```
   hadbm unregisterpackage [--hosts=<host_list>] <old_package_name>
   ```

   Remove the old installation from the file system, as described in the HADB installation instructions (`http://clustra.norway.sun.com/intraweb/download/products/hadb/packages/pdf/4.4.2-6.pdf`).

Known SQL Limitations

- It is not possible to create a `UNIQUE` secondary index on a table.
- The expression `(DISTINCT column)` is not allowed in an aggregate expression, unless this is the only selected expression.
- All tables must be created with a primary key specification (that is, tables without primary keys are not supported).
- `FULL OUTER JOIN` is not supported.
- `IN` subqueries that are table subqueries are not supported; for example:

  ```
  SELECT SNAME FROM S WHERE (S1#,S2#) IN (SELECT S1#,S2# FROM SP
  WHERE P#='P2')
  ```

- Constraints other than `NOT NULL` and `PRIMARY KEY` are not supported.
- It is possible to assign a new owner to a resource. When doing this, however, privileges granted to the current owner are not granted to the new owner.
- Two or more nested `NOT EXISTS` subqueries where each subquery is not (directly) correlated to outer level of queries, is not supported.
- Column privileges are not supported.
- Row value constructors are allowed only in a `VALUES` clause.
- Subqueries are not accepted as value expressions in row value constructors.
- The following data types cannot be used when creating primary keys:
  - `REAL`
  - `FLOAT`
  - `DOUBLE PRECISION`

- DECIMAL
- NUMERIC

High Availability Load Balancing

The Application Server includes load balancing for HTTP, IIOP, and JMS clients; HTTP session failover support; EJB clustering and failover support; highly available EJB timers; distributed transaction recovery; support for rolling application upgrades; and a high availability database for storing the transient state of J2EE applications.

Availability allows for failover protection of Application Server instances in a cluster. If one Application Server instance goes down, another Application Server instance takes over the sessions that were assigned to the unavailable server. Session information is stored in the HADB. HADB supports the persistence of HTTP sessions, Stateful Session Beans, and Single Sign On credentials.

Application Server Product Releases

The Application Server product is delivered in various ways. The following table identifies the product delivered for the various delivery mechanisms:

Application Server Product Release	Delivery Mechanism
Application Server Enterprise Edition component within the Sun Java Enterprise System.	File-based distribution patch installation needed through Sunsolve
Application ServerStandard and Enterprise Edition Standalone product	File-based and Package-based distribution

Compatibility Issues

In the next major release of the Sun Java System Application Server Enterprise Edition the following incompatibilities will be introduced:

- While the HTTP Service will continue using a DNS cache for better performance, monitoring of the DNS cache will not be available.
- The support for HTTP file caching will be revamped, resulting in changes to configuration and monitoring.
- The format for the access log rotation suffix will be changed to the format supported by date and time objects as specified in `http://java.sun.com/j2se/1.5.0/docs/api/java/text/SimpleDateFormat.html`. The default value in this release, "`%YYYY;%MM;%DD;-%hh;h%mm;m%ss;s`," will continue to be supported but no other variations will be supported.

- Any `domain.xml` elements, attributes and properties no longer supported will be flagged as warnings in the server log and in the upgrade log file as having been deprecated.
- The `server.http-service.dns` node will no longer be available in the monitoring view.
- Some of the attributes from the `server.http-service.file-cache` node may be removed. Consequently, any `asadmin` monitoring command trying to access removed attributes from these nodes will fail.

Deploytool

Deploytool will no longer be available. The equivalent functionality is available in the NetBeans IDE. For more information and to plan a migration, please see J2EE 1.4 tutorial for NetBeans 4.1 at `http://www.netbeans.org/kb/`.

Verifier

- Verifier GUI mode (invoked by `verifier -u`) will no longer be available. The equivalent functionality will be available in the NetBeans IDE.
- The default mode for application verification when using verifier tool will change from "Verify J2EE rules" to "Verify J2EE rules and Sun Application Server Configuration Rules." In other words, by default verifier will test whether an application meets J2EE rules and is configured to run on Sun Application Server. The verifier command will have a command-line switch to test an application for J2EE rules only.

Classloader Changes

In the current release, the JAR and directory entries added to `classpath-prefix`, `server-classpath`, and `classpath-suffix` attributes of `domain.xml` (application server configuration file) are available in the JVM system classpath. An application depending on this behavior might be using the following methods from the class `java.lang.ClassLoader` to access classes or other resources from JVM system classpath:

- `getSystemClassLoader()`
- `getSystemResource()`
- `getSystemResourceAsStream()`
- `getSystemResources`

In the next major release, the JAR and directory entries added to `classpath-prefix`, `server-classpath`, and `classpath-suffix` will no longer be available in the JVM system classpath. If an application uses one of the methods mentioned above, Sun strongly recommends using an equivalent method that does not assume that the resources are available in the system classpath. The equivalent methods that do not rely on the JVM system classpath are available in `java.lang.ClassLoader` and should be used when possible; for example:

EXAMPLE 2–1 Old Code

```
java.net.URL url = ClassLoader.getSystemResource
("com/acme/tools/tools.properties");
```

EXAMPLE 2–2 Suggested Change

```
java.net.URL url = this.getClass().getClassLoader().getResource
("com/acme/tools/tools.properties");
```

If it is not possible to change the code, then you may choose to use a new configuration option that will be added in the next release to set JVM system classpath.

Web Service Security Configuration

Security for Web services can be configured using the files `wss-client-config.xml` and `wss-server-config.xml`. Please note that the content and names of these configuration files are unstable and likely to change. The equivalent functionality will continue to be available.

J2EE Support

The Sun Java System Application Server Enterprise Edition 8.2 supports the J2EE 1.4 platform. The following table describes the enhanced APIs available on the J2EE 1.4 platform:

TABLE 2–5 APIs Available on the J2EE 1.4 Platform

API	Description
Components	
Application and Application Client	Implementation of standard deployment descriptors by means of XML schemas
Enterprise JavaBeans (EJB) 2.1	Timer service and EJB Web-service endpoint
Java Servlet 2.4	Web-service endpoint filter
JavaServer Pages (JSP) 2.0 architecture	Expression language and tag library
J2EE Connector Architecture 1.5	Inbound resource adaptor and Java Message Service (JMS) pluggability
Web Services	
Java Web Services Developer Pack 1.5	Integrated toolkit for building, testing and deploying XML applications, Web services, and Web applications

TABLE 2–5 APIs Available on the J2EE 1.4 Platform *(Continued)*

Java API for XML-based Remote Procedure Calls (JAX-RPC) 1.1	Mapping for WSDL and Java technology and support for development of Web-service clients and endpoints
WS-I Basic Profile 1.0	The enabling element for interoperability using WSDL and SOAP
SOAP with attachment API for Java (SAAJ) 1.2	An API for SOAP-based messaging; fosters the creation of SOAP messages with attachments
Java APIs for XML Registries (JAXR) 1.0	A uniform and standard API for accessing XML registries, such as those for Universal Description Discovery and Integration (UDDI and ebXML)
Other	
J2EE Deployment 1.1	Standard APIs that enable deployments of J2EE components and applications
J2EE Management 1.0	Definitions for the information model for managing the J2EE platform
Java Management Extensions (JMX) 1.2	Standard management API
Java Authorization Contract for Containers (JACC) 1.0	Definitions of security contracts between a J2EE Application Server and the authorization policy provider
Java API for XML Processing (JAXP) 1.2	An API with which applications can parse and transform XML documents; also adds support for processing of XML schemas
JMS 1.1	A messaging standard that enables J2EE application components to create, send, receive, and read messages; also adds support for uniform APIs for queues and topics
JavaMail 1.3	A set of abstract classes that model a mail system; also includes minor updates to the APIs

High Performance

The Application Server includes load balancing for HTTP, IIOP, and JMS clients; HTTP session failover support; EJB clustering and failover support; highly available EJB timers; distributed transaction recovery; support for rolling application upgrades; and a high availability database for storing the transient state of J2EE applications.

Availability allows for failover protection of Application Server instances in a cluster. If one Application Server instance goes down, another Application Server instance takes over the sessions that were assigned to the unavailable server. Session information is stored in the HADB. HADB supports the persistence of HTTP sessions, Stateful Session Beans, and Single Sign On credentials.

Scalability

The Application Server supports horizontal scalability through clustering of server instances and request load balancing. It also achieves class leading vertical scalability supporting large multi-processor machines. The integrated message broker can be clustered for better scalability and availability. Client access from HTTP clients, RMI/IIOP based Rich Client Applications, Web Services Clients, and JRM Clients can be load balanced to Application Server clusters.

JavaServer Faces 1.1 Support

The Sun Java System Application Server Enterprise Edition 8.2 supports JavaServer Faces 1.1 technology. The JavaServer Faces technology consists of a set of server-side APIs that represent user-interface components that manage their state, event, handling, and input validation. The APIs also define page navigation and support internationalization and accessibility. You can add custom user-interface components with a JSP custom tag library.

While developing with JavaServer Faces technology, each member of a development team can focus on a single piece of the process. A simple programming model then links the pieces, resulting in a much more efficient and simpler development cycle.

◆ ◆ ◆ CHAPTER 3

Known Issues and Limitations

This chapter describes known problems and associated workarounds for the Sun Java System Application Server Enterprise Edition 8.2 software. If a summary statement does not specify a particular platform, the problem applies to all platforms. This information is organized into the following sections:

Administration

This section describes known administration issues and associated solutions.

Bug ID	Summary
6171458	The `package-appclient` script does not work if `domain1` is not present. By default, there is a hard-coded value in `$INSTALL/lib/package-appclient.xml` for the `AS_ACC_CONFIG` variable for `domain1` that is pointed to by `asenv.conf`. If domain1 is deleted and a new domain created, the `AS_ACC_CONFIG` variable is not updated with the new domain name, which causes the `package-appclient` script to fail. *Solution* Do one of the following: 1. Leave `domain1` intact, and create your other domains around it. 2. Remove `domain1` and replace the hard-coded value for `domain1` in `$INSTALL/lib/package-appclient.xml` with the new domain name. This will have to be done every time a new domain is created if `domain1` is not present.
6196993	Cannot restore backed-up domain with another name. Mirroring of a domain on the same Application Server installation cannot be performed using the `backup-domain` and `restore-domain` commands because the domain cannot be restored using a different name than the original, even though the `asadmin restore-domain` command provides an option to rename the domain. Renaming the backed-up domain appears to succeed, but attempts to start the renamed domain fail because the entries in the domain configuration are not changed, and `startserv` and `stopserv` use the original domain name to set paths. *Solution* The domain name used for `restore-domain` must be the same as that used for the original `backup-domain` command. The `backup-domain` and `restore-domain` commands in Application Server 8.1 work only for backing up and restoring the same domain on the same machine.

Bug ID	Summary

6200011	Starting Application Server with additional JMX Agent is not supported.
	
	J2SE 1.4.x, 5.0, or later can be configured on the Application Server. An integral feature of J2SE 5.0 platform is the ability to start a JMX agent. This is activated when you explicitly set system properties at the server startup.
	
	Example values include:
	
	`name="com.sun.management.jmxremote" value="true"`
	`name="com.sun.management.jmxremote.port" value="9999"`
	`name="com.sun.management.jmxremote.authenticate" value="false"`
	`name="com.sun.management.jmxremote.ssl" value="false"`
	
	After configuring JMX properties and starting the server, a new `jmx-connector` server is started within the Application Server Virtual Machine. An undesirable side-effect of this is that the administration functions are affected adversely, and the Application Server administration Console and command—line interface may produce unexpected results. The problem is that there are some conflicts between the built in `jmx-connector` server and the new `jmx-connector` server.
	
	Solution
	
	If using `jconsole` (or any other `JMX-compliant` client), consider reusing the standard JMX Connector Server that is started with Application Server startup.
	
	When the server starts up, a line similar to the one shown below appears in the server.log. You can connect to the `JMXService` URL specified there and perform the same management/configuration operations after successfully providing the credentials; for example:
	
	`[#|2004-11-24T17:49:08.203-0800|INFO|sun-appserver-ee8.1|javax.enterprise.system.tools.admin|_ThreadID=10;|ADM1501: Here is the JMXServiceURL for the JMXConnectorServer: [service:jmx:rmi:///jndi/rmi://hostname:8686/management/rmi-jmx-connector]. This is where the remote administrative clients should connect using the JSR 160 JMX Connectors.|#]`
	
	For more information, refer to the *Administration Guide*.

6206176	On UNIX, overly restrictive execute permissions on Application Server start and stop scripts. If you run the `asadmin restore-domain` command while logged in as user "A", the scripts will end up with permissions as 744 (`rwxr--r--`). If you subsequently attempt to start or stop a domain as user "B" (even if "B" is root), it will fail because the scripts are only executable for "A". *Solution* Change the permissions on the scripts: `chmod 755` *appserv*`/domains/`*domain-name*`/bin/*`

Bug ID	Summary
6236544, 6275436	Load balancer configuration file does not get created with the endpoint URL of any web service. When setting up the load balancer configuration with an application that has an EJB module that exports a web service URL, the context root for the web service isn't in the resulting `loadbalancer.xml` file. *Solution* 1. Edit the `loadbalancer.xml` file to add the missing web module as follows: `<web-module context-root="context-root-name" disable-timeout-in-minutes="30" enabled="true"/>` 2. Replace *context-root-name* value with the context root name of the web service that was exposed as an EJB.
6288893	Application Server restart using `sun-appserv-admin` causes `LoginException` error. *Solution* 1. Rename the existing <*as_install*>`/bin/asant` script to `asant.bak`. 2. Copy the `asant.template` file in <*as_install*>`/lib/install/templates/ee` (for SE/EE version) to the <*as_install*>`/bin/` directory and rename the file `asant`. 3. Edit the newly copied <*as_install*>`/bin/asant` script, replacing the `%CONFIG_HOME%` token with <*as_install*>`/config`. 4. If there were any manual changes made to the original `asant.bak` file, merge them into the new `asant` script.
6315957	The `.asadmintruststore` file is not described in the Application Server documentation. If this file does not exist in the server administrator's `home` directory, you may experience serious bugs when upgrading certain applications hosted on the server. *Solution* ■ If possible, the `asadmin start-domain domain1` command should be run by user who installed the server. ■ If it is not run by that user, the `.asadmintruststore` should be moved or copied from the `home` directory of installing user to the `home` directory of the running user. ■ Note that if the file is moved (not copied) from the installing user's `home` directory to the running user's `home` directory, you might experience application upgrade problems, as described in bugs 6309079, 6310428 and 6312869, because the upgrade/install user (normally `root` in Java ES) will no longer have the `.asadminstruststore` file in his or her home directory.

Apache and Load Balancer Plugin

This section describes known Apache Web server and load balancer plugin issues and associated solutions.

Bug ID	Summary

6306784	The High-Availability Administration Guide contains incorrect instructions for using `openssl` with Apache.
	Solution
	When compiling and building `openssl`, run the following commands:
	`cd openssl-0.9.7e`
	`config`
	`make`
	Also, for Apache 1.3, the directory name of the `mod_ssl` source will vary depending upon the release of Apache used. For example, for Apache 1.3.33, the name is `mod_ssl-2.8.22-1.3.33`.

6307976	The High-Availability Administration Guide does not contain instructions for using a certificate for Apache 2.0. *Solution* To run Apache security, you must use a certificate. For instructions on obtaining a certificate from a certificate authority, see the information on certificates in the modssl FAQ (`http://www.modssl.org/docs/2.8/ssl_faq.html#ToC24`).

6308021	Must start Apache Web Server as root.
	Solution
	On Solaris, if your Application Server was installed under root, you must start the Apache Web Server as root. Java Enterprise System installations are installed as root. For Apache 2.0, after starting as root, Apache switches and runs as another user you designate. You designate that user in the `/conf/httpd.conf` file. To start as root, on many systems you must edit the `httpd.conf` file to designate the correct group. Replace the line:
	`Group #-1`
	with
	`Group nobody`
	More information on user/group use is included in the `httpd.conf` file.

Bug ID	Summary

6308043	Addition to instructions for using `openssl` with Apache Web Server 2.0 on Solaris.
	After installing Apache 2.0 and the load balancer plug-in, edit `ssl.conf` and `sll-std.conf` as follows:
	Replace the line:
	`<VirtualHost _default_:9191>`
	with
	`<VirtualHost` *machine_name*`:9191>`
	Where *machine_name* is the name of your machine and `9191` is a security port number.

Application Client

This section describes known application client issues and associated solutions.

Bug ID	Summary
6193556	Library JAR packaged in Application Client Archive overwrites MANIFEST file. If you have a top level JAR file inside your client JAR (in this case, reporter.jar), when you deploy the client JAR, the MANIFEST file for that JAR overwrites the MANIFEST file for the client JAR. *Solution* None at this time.

Bundled Sun JDBC Drivers

This section describes known bundled Sun JDBC driver issues and associated solutions.

Bug ID	Summary
6165970	Applications using the `TRANSACTION_SERIALIZABLE` isolation level with the bundled Sun driver for Microsoft SQL Server may hang when using a prepared statement to update if two parallel transactions are running and one of them is rolled back. To set a desired isolation level for a connection, the corresponding connection pool must be created at that same isolation level. See the Administration Guide for details about configuring connection pools. *Solution* None at this time.

6170432	`PreparedStatement` Errors.
	
	Description #1
	
	If an application generates more than 3000 `PreparedStatement` objects in one transaction, the following error may occur with DB2:
	
	`[sunm][DB2 JDBC Driver] No more available statements.Please recreate your package with a larger dynamicSections value.`
	
	Solution #1
	
	Add following properties to the connection pool definition to get the driver to rebind DB2 packages with a larger dynamic sections value:
	
	`createDefaultPackage=true replacePackage=true dynamicSections=1000`
	
	See the *Administration Guide* for details about configuring connection pools.
	
	Description #2
	
	Related to the `PrepardStatement` error above, another error message that may be thrown is:
	
	`[sunm][DB2 JDBC Driver][DB2]Virtual storage or database resource is not available.`
	
	Solution #2
	
	Increase the DB2 server configuration parameter *APPLHEAPSZ*. A good value is 4096.
	
	Description #3
	
	Isolation level `TRANSACTION_SERIALIZABLE`. If your application uses isolation level `TRANSACTION_SERIALIZABLE` and uses one of the parameters suggested above, it might hang while obtaining a connection.
	
	Solution #3
	
	To set desired isolation level for a connection, the corresponding connection pool has to be created at that isolation level. See the *Administration Guide* for instructions.

Bug ID	Summary

6189199	Problems setting isolation level with the bundled Sun driver for Sybase Adaptive Server.
	■ Applications using the `TRANSACTION_SERIALIZABLE` isolation level with the bundled Sun driver for Sybase Adaptive Server may hang when using a prepared statement to update if two parallel transactions are running and one of them is rolled back. Connection rollback fails with following message, and the rolled back connections cannot be used anymore:
	`java.sql.SQLException: [sunm][Sybase JDBC Driver]Request cannot be submitted due to wire contention`
	■ Sybase Adaptive Server does not support the `TRANSACTION_REPEATABLE_READ` isolation level. However, querying `DatabaseMetaData`, the bundled Sun driver returns that this isolation level is supported by the database. Applications using the this isolation level will fail.
	■ Applications using the bundled Sun driver cannot set the `TRANSACTION_READ_UNCOMMITTED` isolation level. The application throws the following exception on the first `DataBaseMetaData` access:
	`java.sql.SQLException: [sunm][Sybase JDBC Driver][Sybase]The optimizer could not find a unique index which it could use to perform an isolation level 0 scan on table 'sybsystemprocs.dbo.spt_server_info'.`
	Solution
	None at this time.

6247468	On Solaris 10 and Enterprise Linux 3.0, the Sun bundled Oracle JDBC driver does not allow the creation of a connection. *Solution* Set the following property on the JDBC connection pool when using the SUN JDBC oracle datasource (`com.sun.sql.jdbcx.oracle.OracleDataSource`): `<property name="serverType" value="dedicated"/>` The value of the property depends upon the way the Oracle server's listener is configured. If it is configured in the "shared" mode, the above value needs to change to "dedicated".

Connectors

This section describes known J2EE connector architecture issues and associated solutions.

Bug ID	Summary
6188343	After restarting a DAS instance, undeploying the connector module fails when cascade is set to false. In this scenario, a standalone or embedded connector module is deployed in DAS and connector connection pools, and resources are created for the deployed module. After restarting the DAS instance, undeploying the connector module fails when cascade is set to false with the following exception: `[#|2004-10-31T19:52:23.049-0800|INFO|sun-appserver-ee8.1|javax.enterprise.system .core|_ThreadID=14;|CORE5023: Error while unloading application [foo]|#]` *Solution* Use cascaded undeploy (set the cascade option to true) for undeploying standalone and embedded connectors after restart of the DAS instance.

Documentation

This section describes known documentation issues and associated solutions.

Bug ID	Summary
Various IDs	Javadoc Inconsistencies. The Javadoc for several AMX interfaces and methods is either missing or incorrect: ■ Getter methods for `NumConnAcquired` and `NumConnReleased` statistics are missing from `ConnectorConnectionPoolStats` and `AltJDBCConnectionPoolStats`. These getter methods will be added in a future release as `getNumConnAcquired()` and `getNumConnReleased()`. ■ Calling the following methods in `EJBCacheStats` will throw an exception: `getPassivationSuccesses()`, `getExpiredSessionsRemoved()`, `getPassivationErrors()`, `getPassivations()`. This will be fixed in a future release. ■ The AMX MBeans may require several seconds after server startup before they are all registered and available for use. A future release will make it possible to determine when the AMX MBeans are fully loaded. ■ The constant `XTypes.CONNNECTOR_CONNECTION_POOL_MONITOR` is misspelled (`"NNN"`). This will be corrected in a future release.
6265624	Bundled ANT throws `java.lang.NoClassDefFoundError`. The following exception is thrown in thread `main`: `java.lang.NoClassDefFoundError: org/apache/tools/ant/launch/Launcher`. *Solution* Using the bundled ANT for activities outside the Application Server is not recommended.

High Availability

This section describes known high availability database (HADB) issues and associated solutions.

Bug ID	Summary
no ID	HADB Configuration with Double Networks. HADB configured with double networks on two subnets works properly on Solaris SPARC. However, due to problems in the operating system or network drivers on some hardware platforms, it has been observed that Solaris x86 and Linux platforms do not always handle double networks properly. This causes the following problems with HADB: ■ On Linux, some of the HADB processes are blocked when sending messages. This causes HADB node restarts and network partitioning. ■ On Solaris x86, some problems may arise after a network failure that prevent switching to the other network interface. This does not happen all the time, so it is still better to have two networks than one. These problems are partially solved in Solaris 10. ■ Trunking is not supported. ■ HADB does not support double networks on Windows 2003 (ID 5103186).
no ID	HADB Database Creation Fails. Creating a new database may fail with the following error, stating that too few shared memory segments are available: `HADB-E-21054: System resource is unavailable: HADB-S-05512: Attaching shared memory segment with key "xxxxx" failed, OS status=24 OS error message: Too many open files.` *Solution* Verify that shared memory is configured and the configuration is working. In particular, on Solaris 8, inspect the file `/etc/system`, and check that the value of the variable `shmsys:shminfo_shmseg` is at least six times the number of nodes per host.
5091280	`hadbm set` does not check resource availability (disk and memory space). When increasing device or buffer sizes using `hadbm set`, the management system checks resource availability when creating databases or adding nodes, but does not check if there are sufficient resources available when device or main-memory buffer sizes are changed. *Solution* Verify that there is enough free disk/memory space on all hosts before increasing any of the `devicesize` or `buffersize` configuration attributes.

Bug ID	Summary

5091349	Heterogeneous paths for `packagepath` not supported.
	
	It is not possible to register the same software package with the same name with different locations at different hosts; for example:
	
	`hadbm registerpackage test --packagepath=/var/install1 --hosts europa11`
	`Package successfully registered.`
	`hadbm registerpackage test --packagepath=/var/install2 --hosts europa12`
	`hadbm:Error 22171: A software package has already been registered with the package name test.`
	
	Solution
	
	HADB does not support heterogeneous paths across nodes in a database cluster. Make sure that the HADB server installation directory (`--packagepath`) is the same across all participating hosts.

6173886, 6253132	`createdomain` may fail. If running the management agent on a host with multiple netwrok interfaces, the createdomain command may fail if not all network interfaces are on the same subnet: `hadbm:Error 22020: The management agents could not establish a domain, please check that the hosts can communicate with UDP multicast.` The management agents will (if not configured otherwise) use the "first" interface for UDP multicasts ("first" as defined by the result from `java.net.NetworkInterface.getNetworkInterfaces()`). *Solution* The best solution is to tell the management agent which subnet to use (set `ma.server.mainternal.interfaces` in the configuration file, e.g., `ma.server.mainternal.interfaces=10.11.100.0`). Alternatively one may configure the router between the subnets to route multicast packets (the management agent uses multicast address 228.8.8.8). Before retrying with a new configuration of the management agents, you may have to clean up the management agent repository. Stop all agents in the domain, and delete all files and directories in the repository directory (identified by `repository.dr.path` in the management agent configuration file). This must be done on all hosts before restarting the agents with a new configuration file.

Bug ID	Summary

6230792, 6230415	Starting, stopping, and reconfiguring HADB may fail or hang.
	
	On Solaris 10 Opteron, starting, stopping or reconfiguring HADB using the `hadbm` command may fail or hang with one of the following errors:
	
	`hadbm:Error 22009: The command issued had no progress in the last 300 seconds.`
	`HADB-E-21070: The operation did not complete within the time limit, but has not been cancelled and may complete at a later time.`
	
	This may happen if there are inconsistencies reading/writing to a file (`nomandevice`) which the `clu_noman_srv` process uses. This problem can be detected by looking for the following messages in the HADB history files:
	
	`n:3 NSUP INF 2005-02-11 18:00:33.844 p:731 Child process noman3 733 does not respond.`
	`n:3 NSUP INF 2005-02-11 18:00:33.844 p:731 Have not heard from it in 104.537454 sec.`
	`n:3 NSUP INF 2005-02-11 18:00:33.844 p:731 Child process noman3 733 did not start.`
	
	Solution
	
	The following workaround is unverified, as the problem has not been reproduced manually. However, running this command for the affected node should solve the problem.
	
	`hadbm restartnode --level=clear` *nodeno dbname*
	
	Note that all devices for the node will be reinitialized. You may have to stop the node before reinitializing it.

6232140	The management agent terminates with the exception "`IPV6_MULTICAST_IF failed`" When starting on a host running Solaris 8 with several NIC cards installed, if there is a mixture of cards with IPv6 and IPv4 enabled, the management agent may terminate with the exception "`IPV6_MULTICAST_IF failed.`" *Solution* Set the environment variable `JAVA_OPTIONS` to `-Djava.net.preferIPv4Stack=true`; for example: `export JAVA_OPTIONS="-Djava.net.preferIPv4Stack=true"` Alternatively, use Solaris 9 or later, which do not exhibit this problem.

Bug ID	Summary
6249685	`clu_trans_srv` cannot be interrupted. There is a bug in the 64-bit version of Red Hat Enterprise Linux 3.0 that makes the `clu_trans_srv` process end up in an uninterruptible mode when performing asynchronous I/O. This means that kill -9 does not work and the operating system must be rebooted. *Solution* Use a 32-bit version of Red Hat Enterprise Linux 3.0.
6262824	`hadbm` does not support passwords containing capital letters. Capital letters in passwords are converted to lowercase when the password is stored in `hadb`. *Solution* Do not use passwords containing capital letters.
6265419	Downgrading from HADB Version 4.4.2.5 to HADB Version 4.4.1.7 causes ma to fail with different error codes. When downgrading to a previous HADB version, the management agent may fail with different error codes. *Solution* It is possible to downgrade the HADB database, however the management agent cannot be downgraded if there changes have been made in the repository objects. After a downgrade, you must keep use the management agent from the latest HADB version.
6271063	Install/removal and `symlink` preservation. Regarding install/removal of HADB c package (Solaris: `SUNWhadbc`, Linux: `sun-hadb-c`) version <m.n.u-p>, the `symlink /opt/SUNWhadb/<m>` is never touched once it exists. Thus, it is possible that an orphaned `symlink` will exist. *Solution* Delete the `symlink` before install or after uninstall unless in use.
6273681	Management agents in global and local zones may interfere. On Solaris 10, stopping a management agent by using the ma-initd script in a global zone stops the management agent in the local zone as well. *Solution* Do not install the management agent both in the global and local zone.

Bug ID	Summary
6275103	`hadbm/ma` should give a better error message when a session object has timed out and deleted at MA. Sometimes, a resource contention problem on the server may cause a management client to become disconnected, When reconnecting, a misleading error message "`hadbm:Error 22184: A password is required to connect to the management agent`" may be returned. *Solution* Check if there is a resource problem on the server, take proper action (e.g., add more resources), and retry the operation.
6275319	Non-root users cannot manage HADB. Installing with Java Enterprise System (as root) does not permit non-root users to manage HADB. *Solution* Always login as root to manage HADB.
6293912	The Management Agent should not use special-use interfaces. Special use interfaces with IP addresses like `0.0.0.0` should not be registered as valid interfaces to be used for HADB nodes in the Management Agent. Registering such interfaces may cause problems if HADB nodes are set up on these interfaces by means of a user issuing a `hadbm create` command using host names instead of IP addresses. The nodes will then be unable to communicate, causing the `create` command to hang. *Solution* When using `hadbm create` on hosts with multiple interfaces, always specify the IP addresses explicitly using DDN notation.
6291562	Reassembly failures on Windows. On the Windows platform, with certain configurations and loads, there may be a large number of reassembly failures in the operating system. The problem has been seen with configurations of more than twenty nodes when running several table scans (`select *`) in parallel. The symptoms may be that transactions abort frequently, repair or recovery may take a long time to complete, and there may be frequent timeouts in various parts of the system. *Solution* To fix the problem, the Windows registry variable `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters` can be set to a value higher than the default `100`. It is recommended that you increase this value to `0x1000` (`4096`). For more information, see. article 811003 (`http://support.microsoft.com/default.aspx?scid=kb;en-us;811003`) from the Microsoft support pages.

Installation

This section describes known installation issues and associated solutions.

Bug ID	Summary
5009728	Installation shutdown hanging on some Linux systems after clicking the "Finish" button. This problem has been observed on several Linux systems. It is most common on Java Desktop System 2 but has also been observed on Linux Red Hat distributions. After clicking the "Finish" button on the last installer screen, the installer fails to launch a browser window containing the product About page or product registration page, and hangs indefinitely, not returning the command prompt. *Solution* Exit the installer by pressing Ctrl+C in the terminal window in which the installer was started. After doing this, browser window containing product About page or registration page will sometimes be launched, but if it does not show up, start the browser and enter following URL in order to review About page: `file://install_dir/docs-ee/about.html` If you also selected the installation option to register the product, follow the link to registration page available on product About page.

6199697	On Windows, the `imq` directory needs to be created during installation.
	
	On Windows, immediately after installing Application Server Enterprise Edition, the Message Queue broker fails on startup with a message saying the directory `drive:\as\domains\domain1\imq` does not exist.
	
	Note that if the broker is started after starting `domain1`, the directory will be created by the Application Server and the problem will not occur.
	
	Solution
	
	1. Create the `var_home_dir_location` before creating the broker:
	
	`$imqbrokerd -varhome` *var_home_dir_location*
	`For example:`
	
	`$imqbrokerd -varhome D:\as\domains\domain1\imq`

6297837	The Application Server installer shows the wrong product release date in the product name, “Sun Java(TM) System Application Server Enterprise Edition 8.1 2005Q4.” *Solution* The correct product name/date should read “Sun Java(TM) System Application Server Enterprise Edition 8.1 2005Q2.”

J2EE Tutorial

To run the J2EE 1.4 Tutorial on the Sun Java System Application Server Enterprise Edition 8.2 perform these tasks:

- When you edit the file examples/`common/build.properties` as described in the "About the Examples" section of the "About this Tutorial" chapter, also change port 4848 to 4849.
- When using Deploytool, add the server `localhost:4849` before deploying an example.
- When using the Administration Console to create any resource, use the Targets tab to specify the server as the target. If you use the command line or an `asant` target, the server is the default target, no further action is required.

Lifecycle Management

This section describes known lifecycle management issues and associated solutions.

Bug ID	Summary

6193449	After setting the `ejb-timer-service` property minimum-delivery-interval to 9000, an attempt to set the `ejb-timer-service` property `redelivery-interval-in-mills` to 7000 causes the set command to fail with the following error:
	
	`[echo] Doing admin task set`
	`[exec] [Attribute(id=redelivery-interval-internal-in-millis) :`
	`Redelivery-Interval (7,000) should be greater than or equal to`
	`Minimum-delivery-interval-in-millis (9,000)]`
	`[exec] CLI137 Command set failed.`
	■ `minimum-delivery-interval` is the minimal interval duration between deliveries of the same periodic timer.
	■ `redelivery-interval-in-mills` is the time the timer service will wait after a failed `ejbTimeout` before attempting redelivery.
	
	The problem is that the logic that relates the redelivery interval property to the minimum delivery property is incorrect and prevents you from using the GUI or the CLI to set any value where the minimum delivery interval is greater than redelivery interval.
	
	The `minimum-delivery-interval-in-millis` must always be set equal to or higher than `ejb-timer-service` property `redelivery-interval-in-millis`. The problem is that there is an erroneous validation check in the Application Server to verify that the value for `redelivery-interval-in-millis` is greater than the value for `minimum-delivery-interval-in-millis`.
	
	Solution
	
	Use the default values for these properties, as follows:
	
	`minimum-delivery-interval(default)=7000`
	`redelivery-interval-in-millis(default)=5000`
	
	Values other than these defaults will generate an error.

Logging

This section describes known logging issues and solutions.

Bug ID	Summary
6180095	Setting debug statement for `access,failure` causes hanging in Application Server startup. Setting the `java.security.debug` option for the JVM will cause the server instance startup to freeze with a deadlock; for example, setting the following in `domain.xml` causes the problem: `<jvm-options>-Djava.security.debug=access,failure</jvm-options>` None at this time. Please avoid setting this flag.

Message Queue

This section describes known Java message queue issues and associated solutions.

Bug ID	Summary
6173308, 6189645, 6198481, 6199510, 6208728	JMS reconnection does not successfully complete in certain cases that are timing dependent. Failures to reconnect in timing-dependent scenarios can be caused by several problems. *Solution* You can work around these problems by: ■ Restarting the brokers involved ■ Restarting the Application Server instances involved
6198465	Asynchronous message listener behavior changed in `appclient` from 8.0 to 8.1 Update 2. Due to a recent change, when an asynchronous message listener is the only live thread in the `app-client` container, the remaining `appclient` virtual machine exists as a daemon. This behavior is a regression for past applications that perform asynchronous receives in ACC. This problem affects application clients that set a JMS message listener and exit the main thread. *Solution* Do not exit the main thread. Wait for the message listener to notify the main thread before terminating the main thread.

Monitoring

This section describes known monitoring issues and associated solutions.

Bug ID	Summary

6174518	Some of the HTTP Service monitoring statistics do not present useful information and should be ignored.
	
	When viewing the monitoring statistics of some elements of the HTTP Service, some values presented do not correspond to current values or are always 0. Specifically, the following HTTP Service statistics do not present information applicable to the Application Server, and should be ignored:
	■ `http-service`
	`load1MinuteAverage`
	`load5MinuteAverage`
	`load15MinuteAverage`
	`rateBytesTransmitted`
	`rateBytesReceived`
	■ `pwc-thread-pool` (the element)
	
	Solution
	
	These monitors will be removed in future releases and replaced with more appropriate information.

Bug ID	Summary

6191092	Monitoring MBean for an undeployed EJB module is not removed, even though all statistics under that monitoring name are moved.
	
	For example:
	
	`EJBModuleMonitorMap().size() = 1  eventhough ejb module is undeployed`
	`EJBModuleMonitor().getName() = sqe_ejb_s1_01`
	
	This true for both EJB modules and applications. Both programmatically (through MBean API) and through `asadmin list/get`, an empty monitoring MBean still exists.
	
	Diagnostics
	
	`asadmin list -m "server.applications" shows the following output:`
	`server.applications.MEjbApp`
	`server.applications.__ejb_container_timer_app`
	`server.applications.adminapp`
	`server.applications.admingui`
	`server.applications.com_sun_web_ui`
	`server.applications._export_install_nov-11_domains_domain1_applications`
	`_j2ee-modules_sqe_ejb_s1_01`
	
	You can look at statistics:
	
	`bin/asadmin list -m "server.applications._export_install_nov-11_`
	`domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01"`
	`server.applications._export_install_nov-11_domains_domain1_`
	`applications_j2ee-modules_sqe_ejb_s1_01.SQEMessageserver.`
	`applications._export_install_nov-11_domains_domain1_applications_`
	`j2ee-modules_sqe_ejb_s1_01.TheGreeter`
	
	Once you undeploy:
	
	`_export_install_nov-11_domains_domain1_applications_j2ee-modules_`
	`sqe_ejb_s1_01`
	
	If you do a list command, you still see the application:
	
	`asadmin list -m "server.applications"`
	`server.applications.MEjbApp`
	`server.applications.__ejb_container_timer_app`
	`server.applications._export_install_nov-11_domains_domain1_`
	`applications_j2ee-modules_sqe_ejb_s1_01`
	`server.applications.adminapp`
	`server.applications.admingui`
	`server.applications.com_sun_web_ui`
	
	but it does not contain any monitoring statistics:
	
	`asadmin list -m "server.applications._expo`
	`rt_install_nov-11_domains_domain1_applications_j2ee-modules_`
	`sqe_ejb_s1_01"`
	`Nothing to list at server.applications.-export-install-nov-`
	`11-domains-domain1-applications-j2ee-modules-sqe-ejb-s1-01.`
	
	To get the valid names beginning with a string, use the wildcard (‘*') character. For example, to list the names beginning with "server.", use list "`server.*`".
	
	Solution

PointBase

This section describes known and associated solutions related to PointBase.

Bug ID	Summary
6184797	Setting the isolation levels on a connection pool for an application causes exceptions in PointBase. For a JDBC connection pool pointing to a PointBase database installation, setting the transaction-isolation-level pool attribute to any value other than the default (`Connection.TRANSACTION_READ_COMMITTED`) causes an exception. However, setting this same parameter to non-default values for pools pointing to other databases does not throw an exception. *Solution* For a JDBC connection pool pointing to a PointBase database installation, do not attempt to set the `transaction-isolation-level`.
6204925	PointBase throws an exception if a network server and embedded drivers are used together. The bundled PointBase sometimes throws an exception if the network server driver and the embedded driver are simultaneously used. *Solution* Use either the embedded driver or the network server driver, but not both.
6264969, 6275448	Upgrade problem where the default PointBase database is overwritten. When upgrading to Application Server Enterprise Edition 8.2 , the Update release patch overwrites the Pointbase default database. *Solution* Recreate or re-enter any scheme or data that existed prior to the upgrade. If you deployed applications with CMP beans with the generate table option, you must undeploy or redeploy the application to have the tables regenerated.

Samples

This section describes known and associated solutions related to the sample code included with the Application Server 8.2 product.

Bug ID	Summary

6195092	On Windows, setup-one-machine-cluster hangs but works on Solaris; `mqfailover` requires `Ctrl+C` to cancel and then must be re-run.
	
	From *install_dir*`\samples\ee-samples\failover\apps\mqfailover\docs\index.html`, if you run the following commands:
	■ Console 1
	`cd` *install_dir*`\samples\ee-samples asant start-mq-master-broker1`
	■ Console 2
	`cd` *install_dir*`\samples\ee-samples asant start-mq-cluster-broker1`
	■ Console 3
	`cd` *install_dir*`\samples\ee-samples asant start-mq-cluster-broker2`
	■ Console 4
	`cd` *install_dir*`\samples\ee-samples asadmin start-domain domain1`
	
	If you have already executed `asant setup-one-machine-cluster-without-ha` or `asant setup-one-machine-cluster-with-ha` for any other Enterprise Edition sample, then execute `asant configure-mq` otherwise execute `asant setup-one-machine-cluster-and-configure-mq`. In this case, the command appears to succeed:
	
	`start_nodeagent: [echo] Start the node agent cluster1-nodeagent`
	`[exec] Command start-node-agent executed successfully.`
	
	But then the system hangs indefinitely.
	
	Solution
	
	None at this time. This problem similarly affects all Enterprise Edition samples that use this ant target on Windows. A workaround is to `Ctrl+C` out of the hung process and then rerun it.

Bug ID	Summary

6198003	Documentation does not explicitly state that you need to create JMS resources before running the MQ Failover Sample Application following the `asadmin` deploy instructions.
	
	The error thrown is as follows:
	
	`/opt/SUNWappserver/domains/domain1/config/sun-acc.xml -name MQFailoverTestClient -textauth -user j2ee -password j2ee`
	`Nov 18, 2004 10:50:17 PM com.sun.enterprise.naming.NamingManagerImpl bindObjects`
	`SEVERE: NAM0006: JMS Destination object not found: jms/durable/TopicA`
	`Nov 18, 2004 10:50:18 PM com.sun.enterprise.naming.NamingManagerImpl bindObjects`
	`SEVERE: javax.naming.NameNotFoundException`
	`javax.naming.NameNotFoundException`
	
	The documentation does not explicitly state that JMS resources must be manually created if manual deployment is done using `asadmin deploy` commands, and that the provided ant targets to deploy the sample application should be used.
	
	Solution
	
	Use the `asant` deploy target for the `build.xml` script, which creates the required JMS resources to run the application.

Bug ID	Summary

6198239	On Linux, a runtime error is displayed during certificate creation in web services/security samples.
	
	When deploying the *install_dir*`/samples/webservices/security` sample (`basicSSl`) on Linux, the certificate is not created and an error similar to the following is thrown:
	
	`generate_certs: [echo] ***Exporting certificate from NSS database`
	`[exec] Result: 1 [echo] ***Generating Java Keystore from generated certificate`
	`[exec] keytool error: java.lang.Exception: Input not an X.509 certificate [exec] Result: 1 [echo] ***Generating Java trust store from generated certificate`
	`[exec] keytool error: java.lang.Exception: Input not an X.509 certificate`
	`[exec] Result: 1`
	`.`
	`.`
	`.`
	`generate_certs: [echo] ***Exporting server certificate from NSS database to a PKCS12 certificate file`
	`[exec] /opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so: version 'NSS_3.9' not found (required by /opt/sun/appserver/lib/pk12util)`
	`[exec] /opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so: version 'NSS_3.6' not found (required by /opt/sun/appserver/lib/pk12util)`
	`[exec] /opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so: version 'NSS_3.7' not found (required by /opt/sun/appserver/lib/pk12util)`
	`[exec] Result: 1`
	
	The problem is that NSS libraries are in different locations on Linux installations than on Solaris installations. You need to make sure that the `LD_LIBRARY_PATH` points to the proper NSS libraries when deploying on Linux. Either set `LD_LIBRARY_PATH` in your environment, or set it in the *install_dir*`/bin/asant` shell wrapper script.
	
	Solution
	
	Do one of the following:
	■ Set `LD_LIBRARY_PATH=/opt/sun/private/lib`.
	■ Add to the following line to the *install_dir*`/bin/asant` script:
	
	`LD_LIBRARY_PATH=$AS_NSS:$LD_LIBRARY_PATH;export LD_LIBRARY_PATH`

Security

This section describes known issues and associated solutions related to Application Server and web application security and certificates.

Bug ID	Summary
6183318	Cannot run `WebServiceSecurity` applications on Enterprise Edition with J2SE 5.0. `WebServiceSecurity` applications cannot run with J2SE 5.0 because: ■ J2SE 5.0 PKCS11 does not support UNWRAP mode ■ J2SE 5.0 PKCS11 does not support `RSA/ECB/OAEPWithSHA1AndMGF1Padding` with PKCS11 The J2SE team has filed "CR 6190389: Add support for the RSA-PKCS1 and RSA-OAEP wrap/unwrap mechanisms" for this bug. *Solution* Use J2SE 1.4.2 with any other JCE provider (not the one included by default). Note that hardware accelerator support will not be present in this configuration.
6269102	SSL termination is not working; when Load Balancer (Hardware) is configured for SSL termination, the Application Server changes the protocol from `https` to `http` during redirection. *Solution* Add a software load balancer between the hardware load balancer and the Application Server.

Upgrade Utility

This section describes known Upgrade utility issues and associated solutions.

Bug ID	Summary
6165528	Domains created in custom-path other than *install_dir*`/domains` directory are not upgraded directly while upgrading from Application Server Enterprise Edition 8 to Application Server Enterprise Edition 8.1. When running the Upgrade Utility and identifying the *install_dir* as the source installation directory, the upgrade process upgrades only those domains that are created under *install_dir*`/domains` directory. Domains created in other locations are not upgraded. *Solution* Before starting the upgrade process, copy all the domain directories from their different locations to the *install_dir*`/domains` directory.

Bug ID	Summary
6207337	On some Linux systems, the installer running "Upgrade in place" fails to start upgrade tool after clicking on the "Start Upgrade Wizard" button. This problem has been observed on several Linux systems, it is most common on Java Desktop System 2 but has also been observed on Red Hat distributions. After clicking the "Start Upgrade Tool" button on the final installer screen, the installer fails to launch the upgrade tool to complete the upgrade process, and hangs indefinitely, not returning the command prompt. *Solution* This issue is not encountered if command line installation mode is used to run upgrade in place. 1. If you ran upgrade in place in GUI mode and encountered this problem, exit the installer by pressing Ctrl+C in the terminal window in which the installer was started. 2. Start upgrade tool from the terminal window, using following command: *install_dir*`/bin/asupgrade --source install_dir/domains --target install_dir --adminuser adminuser --adminpassword adminpassword --masterpassword changeit` *adminuser* and *adminpassword* `should match the values used for the installation you are upgrading.` 3. When the upgrade tool completes the upgrade process you can also start the browser and enter following URL in order to review About page: `file://`*install_dir*`/docs-ee/about.html` If you also selected the installation option to register the product, follow the link to registration page available on product About page.
6296105	Self-signed certificate is not trusted during and after upgrade from 8.0 Platform Edition (PE) to 8.1 Enterprise Edition (EE) UR2. *Solution* Remove the following entries from the target `domain.xml` (after the upgrade) and restart the server: `<jvm-options>-Djavax.net.ssl.keyStore=${com.sun.aas.instanceRoot}/config/keystore.jks</jvm-options>-<jvm-options>Djavax.net.ssl.trustStore=${com.sun.aas.instanceRoot}/config/cacerts.jks</jvm-options>`

Web Container

This section describes known web container issues and associated solutions.

Bug ID	Summary
5004315	On Windows, deploying an application using `--precompilejsp=true` can lock JAR files in the application, causing later undeployment or redeployment to fail. If you request precompilation of JSPs when you deploy an application on Windows, later attempts to undeploy that application or to redeploy it (or any application with the same module ID) will not work as expected. The problem is that JSP precompilation opens JAR files in your application but does not close them, and Windows prevents the undeployment from deleting those files or the redeployment from overwriting them. Note that undeployment succeeds to a point, in that the application is logically removed from the Application Server. Also note that no error message is returned by the `asadmin` utility, but the application's directory and the locked jar files remain on the server. The server's log file will contain messages describing the failure to delete the files and the application's directory. Attempts to redeploy the application after undeploying fail because the server tries to remove the existing files and directory, and these attempts also fail. This can happen if you try to deploy any application that uses the same module ID as the originally deployed application, because the server uses the module ID in choosing a directory name to hold the application's files. Attempts to redeploy the application without undeploying it first will fail for the same reasons. *Diagnostics* If you attempt to redeploy the application or deploy it after undeploying it, the asadmin utility returns an error similar to the one below. `An exception occurred while running the command. The exception message is: CLI171 Command deploy failed : Deploying application in domain failed; Cannot deploy. Module directory is locked and can’t be deleted.` *Solution* If you specify `--precompilejsps=false` (the default setting) when you deploy an application, then this problem will not occur. Be aware that the first use of the application will trigger the JSP compilation, so the response time to the first request will be longer than for later requests. Note also that if you do precompile, you should stop and restart the server before undeploying or redeploying the application. The shutdown frees the locked JAR files so the undeployment or redeployment after the restart can succeed.

Bug ID	Summary
6172006	Unable to deploy WAR with Servlet 2.4-based `web.xml` that contains an empty `<load-on-startup>` element. The optional load-on-startup servlet element in a `web.xml` indicates that the associated servlet is to be loaded and initialized as part of the startup of the web application that declares it. The optional content of this element is an integer indicating the order in which the servlet is to be loaded and initialized with respect to the web application's other servlets. An empty `<load-on-startup>` indicates that the order is irrelevant, as long as the servlet is loaded and initialized during the startup of its containing web application. The Servlet 2.4 schema for `web.xml` no longer supports an empty `<load-on-startup>`, meaning that an integer must be specified when using a Servlet 2.4 based `web.xml`. If specifying an empty `<load-on-startup>`, as in `<load-on-startup/>`, the `web.xml` will fail validation against the Servlet 2.4 schema for `web.xml`, causing deployment of the web application to fail. Backwards compatibility issue. Specifying an empty <load-on-startup> still works with Servlet 2.3 based `web.xml`. *Solution* Specify `<load-on-startup>0</load-on-startup>` when using a Servlet 2.4 based `web.xml` to indicate that servlet load order does not matter.

Bug ID	Summary
6184122	Unable to compile JSP page on resource constrained servers. The JSP page is accessed but fails to compile, and the server log contains the error message "Unable to execute command" with the following stack trace: `at org.apache.tools.ant.taskdefs.Execute$Java13CommandLauncher.exec (Execute.java:655) at org.apache.tools.ant.taskdefs.Execute.launch (Execute.java:416) at org.apache.tools.ant.taskdefs.Execute.execute (Execute.java:427) at org.apache.tools.ant.taskdefs.compilers. DefaultCompilerAdapter.executeExternalCompile(DefaultCompilerAdapter. java:448) at org.apache.tools.ant.taskdefs.compilers.JavacExternal. execute(JavacExternal.java:81) at org.apache.tools.ant.taskdefs. Javac.compile(Javac.java:842) at org.apache.tools.ant.taskdefs.Javac. execute(Javac.java:682) at org.apache.jasper.compiler.Compiler. generateClass(Compiler.java:396)`  *Solution*  Set the JSP compilation switch "fork" to "false."  This can be done either of two ways: ■ Globally, by setting the fork init parameter of the JspServlet in `${S1AS_HOME}/domains/domain1/config/default-web.xml` to false:  `<servlet> <servlet-name>jsp</servlet-name> <servlet-class>org.apache.jasper.servlet.JspServlet</servlet-class> .... <init-param> <param-name>fork</param-name> <param-value>false</param-value> </init-param> .... </servlet>`  ■ On a per-web application basis, by setting the fork JSP configuration property in `sun-web.xml` to false:  `<sun-web-app> <jsp-config> <property name="fork" value="false" /> </jsp-config> </sun-web-app>`  Either setting will prevent ant from spawning a new process for `javac` compilation.

Bug ID	Summary
6188932	Application Server does not support `auth-passthrough` Web Server 6.1 Add-On. The Sun Java System Application Server Enterprise Edition 8.2 adds support for the functionality provided by the `auth-passthrough` plugin function available with Sun Java System Application Server Enterprise Edition 7.1. However, in Application Server Enterprise Edition 8.2 , the `auth-passthrough` plugin feature is configured differently. The `auth-passthrough` plugin function in Application Server Enterprise Edition 7.1 has been useful in two-tier deployment scenarios, where: ■ Application Server instance is protected by a second firewall behind the corporate firewall. ■ No client connections are permitted directly to the Application Server instance. In such network architectures, a client connects to a front-end web server, which has been configured with the `service-passthrough` plugin function and forwards HTTP requests to the proxied Application Server instance for processing. The Application Server instance can only receive requests from the web server proxy, but never directly from any client hosts. As a result of this, any applications deployed on the proxied Application Server instance that query for client information, such as the client's IP address, will receive the proxy host IP, since that is the actual originating host of the relayed request. In Application Server Enterprise Edition 7.1, the `auth-passthrough` plugin function could be configured on the proxied Application Server instance in order to make the remote client's information directly available to any applications deployed on it; as if the proxied Application Server instance had received the request directly, instead of via an intermediate web server running the `service-passthrough` plugin. In Application Server Enterprise Edition 8.2 , the `auth-passthrough` feature may be enabled by setting the `authPassthroughEnabled` property of the `<http-service>` element in `domain.xml` to TRUE, as follows: `<property name="authPassthroughEnabled" value="true"/>` The same security considerations of the `auth-passthrough` plugin function in Application Server Enterprise Edition 7.1 also apply to the `authPassthroughEnabled` property in Application Server Enterprise Edition 8.2 . Since `authPassthroughEnabled` makes it possible to override information that may be used for authentication purposes (such as the IP address from which the request originated, or the SSL client certificate), it is essential that only trusted clients or servers be allowed to connect to an Application Server Enterprise Edition 8.2 instance with `authPassthroughEnabled` set to TRUE. As a precautionary measure, it is recommended that only servers behind the corporate firewall should be configured with `authPassthroughEnabled` set to TRUE. A server that is accessible through the Internet must never be configured with `authPassthroughEnabled` set to TRUE. Notice that in the scenario where a proxy web server has been configured with the `service-passthrough` plugin and forwards requests to an Application Server 8.1 Update 2 instance with `authPassthroughEnabled` set to TRUE, SSL client authentication may be enabled on the web server proxy, and disabled on the proxied Application Server 8.1 Update 2 instance. In this case, the proxied Application Server 8.1 Update 2 instance will still treat the request as though it was authenticated via SSL, and provide the client's SSL certificate to any deployed applications requesting it.