BEA WebLogic Server™

Supported Configurations for WebLogic Server 5.1

Version 5.1
Revised: November 19, 2003

Copyright

Restricted Rights Legend

Trademarks or Service Marks

Contents

About This Document

1. Supported Configurations Overview

14.Sun Microsystems/Fujitsu SPARC with Solaris 2.6

15.Sun Microsystems/Fujitsu SPARC with Solaris 2.7

17.Installation Help for Your Platform

18.Errata

About This Document

This document provides information on BEA-supported configurations for WebLogic Server.

Audience

This document is written for all users of WebLogic Server.

e-docs Web Site

BEA product documentation is available on the BEA corporate Web site. From the BEA Home page, click on Product Documentation or go directly to the WebLogic Server Product Documentation page at http://e-docs.bea.com.

How to Print the Document

You can print a copy of this document from a Web browser, one main topic at a time, by using the File—Print option on your Web browser.

A PDF version of this document is available on the WebLogic Server documentation Home page on the e-docs Web site (and also on the documentation CD). You can open the PDF in Adobe Acrobat Reader and print the entire document (or a portion of it) in book format. To access the PDFs, open the WebLogic Server documentation Home page, click Download Documentation, and select the document you want to print.

Adobe Acrobat Reader is available at no charge from the Adobe Web site at http://www.adobe.com.

Related Information

The BEA corporate Web site provides all documentation for WebLogic Server.

For more information about BEA-supported configurations for specific WebLogic Server releases, refer to the following sources:

- Supported Configurations for WebLogic Server
- Supported Configurations for WebLogic Server 8.1
- Supported Configurations for WebLogic Server 7.0
- Supported Configurations for WebLogic Server 6.1

Contact Us!

Your feedback on BEA documentation is important to us. Send us e-mail at docsupport@bea.com if you have questions or comments. Your comments will be reviewed directly by the BEA professionals who create and update the documentation.

In your e-mail message, please indicate the software name and version you are using, as well as the title and document date of your documentation. If you have any questions about this version of BEA WebLogic Server, or if you have problems installing and running BEA WebLogic Server, contact BEA Customer Support through BEA WebSupport at http://www.bea.com. You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages

Documentation Conventions

The following documentation conventions are used throughout this document.

Convention	Usage
Ctrl+Tab	Keys you press simultaneously.
italics	Emphasis and book titles.

`monospace text`	Code samples, commands and their options, Java classes, data types, directories, and file names and their extensions. Monospace text also indicates text that you enter from the keyboard.
	Examples:
	`import java.util.Enumeration;`
	`chmod u+w *`
	`config/examples/applications`
	`.java`
	`config.xml`
	`float`

`monospace italic text`	Variables in code. *Example*: `String CustomerName;`
UPPERCASE TEXT	Device names, environment variables, and logical operators. *Example*s: LPT1 BEA_HOME OR
{ }	A set of choices in a syntax line.
[ ]	Optional items in a syntax line. *Example*: `java utils.MulticastTest -n name -a address [-p portnumber] [-t timeout] [-s send]`
|	Separates mutually exclusive choices in a syntax line. *Example*: `java weblogic.deploy [list|deploy|undeploy|update] password {application} {source}`

Convention	Usage
. . .	Indicates one of the following in a command line: • An argument can be repeated several times in the command line. • The statement omits additional optional arguments. • You can enter additional parameters, values, or other information
. . .	Indicates the omission of items from a code example or from a syntax line.

CHAPTER 1

Supported Configurations Overview

The following sections provide an overview of supported WebLogic Server 5.1 configurations.

List of Supported Platforms

Click a platform name to view supported configuration information for that platform.

- HP Alpha with Tru64 UNIX
- Hewlett-Packard HP/9000 with HP-UX 11.0 and 11i
- IBM AS/400e with OS/400 V4R4/V4R5
- IBM Dynix/ptx
- IBM S/390 with OS/390
- Windows 2000 Professional for IA-32
- Windows 2000 Server, Windows 2000 Advanced Server for IA-32
- Windows NT 4.0 for IA-32
- Red Hat Linux for IA-32
- Sun Microsystems/Fujitsu SPARC with Solaris 2.5.1
- Sun Microsystems/Fujitsu SPARC with Solaris 2.6
- Sun Microsystems/Fujitsu SPARC with Solaris 2.7
- Sun Microsystems/Fujitsu SPARC with Solaris 8

BEA provides full support for these platforms only. We are continuously working to provide support for more platforms. Please contact your sales representative for information about platforms not listed in this table.

End-of-Life and Product Lifecycle Policy Information

For product End-of-Life (EOL) information, see WebLogic Server and WebLogic jDriver End-of-Life Information.

For BEA's product lifecycle policy, see BEA Product Lifecycle Policy.

Download and Installation Information

The following sections provide information on:

- "Download WebLogic Server for Your Platform" on page 1-2
- "Installation Instructions for Your Platform" on page 1-3

Download WebLogic Server for Your Platform

You can download WebLogic Server from one of the following sites:

- Our newest products are featured at http://commerce.bea.com/index.jsp.
- Service Packs, Previous Releases, and Patches for Products are located at http://commerce.bea.com/support/support_mainpage.jsp. You need a valid contract support account ID to access this site.

Installation Instructions for Your Platform

Because setup, download and installation of WebLogic Server are more specialized on some platforms, we have provided additional information on these topics in Installation Help for Your Platform.

For general information on WebLogic Server installation, see Installing and Setting up WebLogic Server 5.1.

How BEA Supports WebLogic Server

BEA supports WebLogic Server on multiple platforms, JVMs, and operating system configurations. BEA validates the operation of supported software on various sub-platforms with rigorous internal testing. In some cases, there are problems with certain JDKs, operating systems, and hardware platforms that prevent BEA from supporting WebLogic Server. BEA updates this site frequently to provide you with the latest information on supported sub-platforms and recommends that you confirm support for your configuration with other vendors you use.

We test our software in a variety of Java environments. Our multi-platform testing program contributes to the high quality of our software by uncovering bugs that only appear in some Java environments. We fix the bugs we find in WebLogic Server, but sometimes we uncover bugs in a particular Java environment or in the underlying operating system. When we uncover bugs such as these, we report the bug to the vendor and work towards a solution. We may also attempt to work around the bug in our software. For some sub-platforms, we specify a particular version of the Java Virtual Machine (JVM) or operating system patches needed to run WebLogic Server.

Development Platforms

WebLogic Server supports Development Platforms for the design, development, and verification of applications. Development Platforms are not supported for production server deployments. We indicate that a platform is supported as a Development Platform by specifically marking a platform as Development Platform Only.

Mixed Client/Server JVMs

BEA specifically supports Java (RMI, not HTTP) clients to run with the same version of the JVM as is used by the WebLogic Server. Deployments using different JVM versions for the client and server are not supported, except for those listed in the following table:

Table 1-1 Support Mixed Client/Server JVMs

Deployment	Details
Windows NT on WebLogic 5.1 with JavaSoft JDK 1.2.1	On Windows NT with WebLogic 5.1, BEA has successfully tested client applications running under JDK 1.1.7 accessing a WebLogic Server running with JavaSoft JDK 1.2.1 and JDK 1.2.2. Note that if you use the Microsoft SDK for Java, you ***must*** run both WebLogic Server and RMI clients with the Microsoft JVM. It is not possible to run RMI clients under the Microsoft SDK against a WebLogic Server running with a non-Microsoft JVM.
Solaris 2.7 on WebLogic 5.1 with Service Pack 9 or higher and JDK 1.2.2_07	Using Solaris 2.7 on WebLogic 5.1.0 with Service Pack 9 or higher, BEA has successfully tested client applications running under JDK 1.3.0 accessing a WebLogic Server running with JDK 1.2.2_07.
Solaris 2.6, 2.7 and 8 on WebLogic 5.1 with Service Pack 9 or higher and SDK 1.3.0	Using Solaris 2.6, 2.7 and 8 on WebLogic 5.1.0 with Service Pack 9 or higher, BEA has successfully tested client applications running under JDK 1.2.2_05a when connecting to a server running SDK 1.3.0.

BEA continues to test new Java environments and platforms. Check this site often for the most current information.

Platform Support for WebLogic jDriver JDBC Drivers

Platform support for the WebLogic jDrivers is described by individual platform. If you use a driver in conjunction with WebLogic Server, see the List of Supported Platforms for information on supported JVMs.

Note: The Type 4 drivers for Microsoft SQL Server and Informix Dynamic Server are pure Java drivers, are not tied to a particular OS and do not require any native libraries or DLLs.

Other Third-Party Drivers

Also note that other third-party JDBC Drivers may be used in conjunction with WebLogic Server. WebLogic is a completely java-based server and is committed to the J2EE standards. As such, WebLogic supports the use of any third-party JDBC driver to any DBMS while using WebLogic Server.

However, there are some limitations. The driver **must** be threadsafe. WebLogic is a highly multithreaded application and there are drivers (the JDBC-ODBC bridge from Sun, for example) which cannot be used with WebLogic. Also, when used in transactionally aware environments, such as for EJBs etc., the driver must implement the standard JDBC transactional calls, such as setAutoCommit(), setTransactionIsolation() etc. Third-party JDBC drivers that do not implement Serializable or Remote interfaces cannot pass objects to a T3 client application.

WebLogic Support of Oracle Versions

The following sections provide information on WebLogic Server platform support for the following Oracle versions :

- Oracle 8.1.6 and earlier versions
- Oracle Server 9.2.0.x

Oracle 8.1.6 and earlier versions

WebLogic Server no longer supports Oracle 8.1.6 and earlier versions; Oracle has dropped support for these versions and is no longer fixing bugs in these products. The.dll files for their use are still available in earlier version of WebLogic, however future releases will no longer include them. If you are doing development work, it is highly recommended to move to 8.1.7 or higher.

Oracle Server 9.2.0.x

WebLogic Server applications that use distributed transactions with Oracle Server 9.2.0.x are experiencing problems with the Oracle thin driver 9.2.0.x. For such applications, BEA recommends against using Oracle thin driver 9.2.0.x at this time. Instead, use a 9.0.1.X driver, where X is 3 or higher.

In addition to the aforementioned workaround, the following tuning technique may help alleviate the problem:

- Set the number of connections in the JDBC configured connection pools to be equal to the number of default execute threads helps. This means WebLogic Server does not have to wait for a connection to complete activity on an XA resource.

More Information on jDrivers

For more information, please see WebLogic 5.1 JDBC Options

Platform Support for WebLogic Plug-ins and Web Servers

Platform support for the WebLogic plug-ins and web servers is described by individual platform. If you use a driver in conjunction with WebLogic Server, see the List of Supported Platforms above for more information.

About Apache Plug-in Support

The Apache plug-in versions listed as supported on a particular platform are available from the Apache Server Foundation. Other products derived from the open source version of Apache, such as Covalent's Enterprise Ready Server, may also work with the Apache plug-in provided with WebLogic Server but have not been tested by BEA. All support testing for products based on open source Apache will be performed by the vendor offering such products.

Platform Support for Jolt

Jolt is a Java-based client API that manages requests to BEA Tuxedo services via a Jolt Service Listener (JSL) running on the Tuxedo server. For more information on Tuxedo and Jolt, see BEA Tuxedo Documentation.

For this release, Jolt 1.2 is bundled with WebLogic Server.

- Jolt 1.2
- Jolt 1.2.1

Platform Support for WebLogic Enterprise Connectivity

WebLogic Enterprise Connectivity (WLEC) is an optional component of WebLogic Server (WLS) version 5.1 and later. Using WLEC, you can execute objects and EJBs in WebLogic

Enterprise™ (WLE) 5.1, Tuxedo 8.0, or Tuxedo 8.1 domains from WLS applications. You enable WLEC by creating an Internet Inter-ORB Protocol (IIOP) connection pool and then writing server-side Java code to use the connection pool.

The next section summarizes WLEC support.

WebLogic Server 5.1

- Solaris 2.6 or 2.7, Windows NT 4.0, HP-UX 11.0 and 11i, AIX 4.3.3, Windows 2000, or Compaq Tru64 Unix 4.
- JDK 1.2.2
- WebLogic Enterprise 5.1

Browser support for WebLogic Server

Note: If you use a browser version that is not listed as a supported browser in the following sections, you may experience functional or formatting problems.

Browser support for WebLogic Server is dependent on the browser version, the client operating system for console, and the Java plug-in for applet support. The following sections summarize browser support for WebLogic Server:

- Browser Support for the WebLogic Server Console
- Browser Support for Applets with WebLogic Server

Browser Support for the WebLogic Server Console

The following table summarizes browser support for the WebLogic Server console.

Client Operating System for Console	Browser version
Intel Pentium-compatible with Windows 2000 Server or Windows 2000 Advanced Server or Intel Pentium-compatible with Windows 2000 Professional or Intel Pentium-compatible with Windows NT 4.0	Netscape 4.7 Internet Explorer 5.x
Sun Microsystems/Fujitsu SPARC with Solaris 2.5.1, 2.6, 2.7	Netscape 4.7x
Sun Microsystems/Fujitsu SPARC with Solaris 8	Netscape 4.7x

Browser Support for Applets with WebLogic Server

The following table summarizes browser support for applets with WebLogic Server.

Netscape version	Internet Explorer version	Sun Java Plug-in
Netscape 4.51, 4.7	Internet Explorer 4.0, 5.0	1.2.2

CHAPTER 2

Bull/IBM pSeries with AIX 4.3.3

General Information

- WebLogic Server will end support for Bull/IBM pSeries with AIX 4.3.3 starting December 31, 2003. For information, see WebLogic Server and WebLogic jDriver End-of-Life Announcements.
- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- AIX is Bull/IBM's version of UNIX for pSeries servers. For information on AIX, go to IBM's website.
- Get the JDK for AIX 4.3.3 and the latest JDK patch from the IBM Java Technology web page.
- `UDP_SENDSPACE` buffer size defaults to 9216 (9k). WebLogic Server sends fragments up to 32K. Using a buffer size smaller than 32k may cause an IO Exception. Users should set the buffer size to 32 k. Use the following command:

  ```
  no -o udp_sendspace=32768
  ```

  Users will need to issue this command and then reboot WebLogic Server.

Supported Configuratons

This section lists the releases supported for this platform:

- WebLogic 5.1

WebLogic 5.1

Operating System	AIX 4.3.x	
Required OS Patches	None.	
CPU	pSeries PowerPC, POWER3 200MHz or higher	
RAM	64 MB Minimum	
Disk Space	77 MB	
Supported Client JVM	• JDK 1.1.8 (AIX 4.3.2 and AIX 4.3.3) • JDK 1.2.2 (AIX 4.3.3) • IBM's JDK 1.2.2 PTF 11 • IBM's JDK 1.3.0	
Supported Server JVM	• JDK 1.1.8 (AIX 4.3.2 and AIX 4.3.3) • JDK 1.2.2 (AIX 4.3.3) • IBM's JDK 1.2.2 PTF 11 • IBM's JDK 1.3.0	
Performance Pack	Included.	`weblogic/lib/aix/libmuxer.so`
Node Manager	None.	

Oracle jDriver	**For this Oracle Server version:**
	Oracle 8.1.7 (Service Pack 9 or higher)
		These Oracle jDriver shared libraries are supported:
		`lib/aix/oci817_8/libweblogicoci37.so`
		`lib/aix/oci817_8/libweblogicoxa37.so`

NSAPI Plug-ins	**For this plugin:**	**This shared library is supported:**
	IPlanet 3.6	`weblogic/lib/aix/libproxy36.so`
	IPlanet 4.0	`weblogic/lib/aix/libproxy40.so`
Apache Plug-ins	**For this Apache version:**	**These shared libraries are supported:**

	Apache 1.3.x	`weblogic/lib/hpux11/mod_wl.so`
		(Use with standard Apache (non EAPI). It is available beginning with WLS 5.1.0 Service Pack 6)
		
		`weblogic/lib/hpux11/mod_wl_ssl.so`
		(The file `mod_wl_ssl.so` is used for Apache + SSL/EAPI installations (Stronghold, modssl etc.). It is available beginning with WLS 5.1.0 Service Pack 6)

Known Issues for WebLogic 5.1

- The `StartWeblogic.sh` script. exports `LIB_PATH` instead of `LIBPATH`. Users should update the script by replacing `export LIB_PATH` with `export LIBPATH as shown below:`

```
case `uname -s` in
AIX)
  if [ -n "$LIBPATH" ]; then
    LIBPATH=$LIBPATH:$WL_HOME/lib/aix
  else
    LIBPATH=$WL_HOME/lib/aix
  fi
  export LIBPATH
  echo "LIBPATH=$LIBPATH"
```

- There are some known issues in using RMI-IIOP with WebLogic 5.1 with JDK 1.2.2.11 as the RMI-IIOP implementation relies on some JDK specific features. BEA addressed these in subsequent releases of Weblogic.

- On AIX, the WebLogic-to-Netscape-Enterprise-Server Bridge (NSAPI) used to synchronize communications from the proxy to the WebLogic Server to avoid problems encountered with these products on AIX. As a result NSAPI on AIX was single-threaded, allowing just one access at a time.
- These problems have been solved with Service Pack 8 for WebLogic 5.1—if you have installed the latest service pack, the NSAPI plug-in is no longer single-threaded.
- IBM highly recommends installing the following APAR's (patches) for AIX users with JDK 1.3.0 that are experiencing memory corruption problem:
  APAR IY27684 (kernel fix)
  IY27836 (malloc fix).
  These, along with Java APAR IY28205(JDK130 PTF 9b), solve issues that required the use of -Xgcthreads.

CHAPTER 3

Compaq OpenVM

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- For general information about Java technology for Compaq Alpha Systems, see the Compaq Java Technology Center website.

Supported Releases

This section lists the releases supported for this platform:

- WebLogic 5.1

WebLogic 5.1

Operating System	Compaq Alpha Server with OpenVMS 7.2
Required OS Patches	None
CPU	433 MHz Alpha
RAM	256 MB minimum
Disk Space	67 MB
Supported Client JVM	JDK 1.2.2-3
Supported Server JVM	JDK 1.2.2-3
Performance Pack	None
Node Manager	None
Oracle jDriver	None
NSAPI Plug-ins	None
Apache Plug-ins	None

Known Issues

- OpenVMS v7.2 has been tested on WebLogic Server 5.1.with jdbcKona/MSSQLServer4. Note the following, however:
  - WebLogic Server must be started statically for the OpenVMS platform.
  - Make sure `compileCommand=javac` is valued under your weblogic.httpd.register.*.jsp entry in the weblogic properties file. This will enable compilation of JSP's on the OpenVMS platform.
- `ActivatableRuntimeDescriptor.getStub()` is not supported.

CHAPTER 4

HP Alpha with Tru64 UNIX

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- For general information about HP Alpha and HP Tru64 UNIX (formerly Compaq Tru64 UNIX), see the HP Tru64 UNIX website.

Supported Releases

This section lists the releases supported for this platform:

- WebLogic 5.1 SP13
- WebLogic 5.1.0

WebLogic 5.1 SP13

Operating System	Tru64 UNIX V5.1A	
CPU	433 MHz Alpha	
Patches	N/A	
RAM	Tested with 2G	
Disk Space	141 MB after installation	
Supported Client JVMs	JDK 1.3.1 Java™ 2 Runtime Environment, Standard Edition Fast VM (build 1.3.1-5, native threads, mixed mode, precompiled rt.jar, 12/16/2002-19:04)	
Supported Server JVMs	JDK 1.3.1 Java™ 2 Runtime Environment, Standard Edition Fast VM (build 1.3.1-5, native threads, mixed mode, precompiled rt.jar, 12/16/2002-19:04)	
Performance Pack	Included.	`lib/tru64unix/libmuxer.so`
Node Manager	None.	
Oracle jDriver	**For this Oracle Server version:** Oracle Server 8.1.7	**This Oracle jDriver shared library is supported:** `lib/tru64unix/oci817_8/libweblogicoci36.so`
NSAPI Plug-ins **For this plugin:**	None.	
Apache Plug-ins	**For these Apache versions:** Secure Web Server (SWS) 5.9.2.	**This shared library is supported:** `lib/tru64unix/mod_wl.so`

Installation Instructions

Use the following instructions to install WebLogic Server 5.1 SP13:

1. Go to the BEA Download Center at http://commerce.bea.com/downloads/weblogic_server.jsp.
2. In the WebLogic Server 5.1.0 with Service Pack 13 drop-down box, select For non-win32 Users.
3. Click Proceed to Download and follow the instructions provided.
4. Users requiring an Apache Plug-in should contact their BEA Support Representative and request the `CR094520_510sp13.jar` file. Extract the file into your *`WL_HOME`*`/lib` directory.

WebLogic 5.1.0

Operating System	Tru64 Unix 4.0b-5.1	
CPU	433 MHz Alpha	
Patches	N/A	
RAM	64 MB	
Disk Space	67 MB	
Supported Client JVMs	JDK 1.1.8_5 JDK 1.2.2 1.2.2:04 (FastVM) Requires WebLogic Service Pack 6; invoke FastVM with `-fast` on the java command line	
Supported Server JVMs	JDK 1.1.8_5 JDK 1.2.2 1.2.2:04 (FastVM) Requires WebLogic Service Pack 6; invoke FastVM with `-fast` on the java command line	
Performance Pack	None.	
Node Manager	None.	

Oracle jDriver	**For this Oracle Server version:**
	Oracle Server 8.1.7
		These Oracle jDriver shared libraries are supported:
		`lib/tru64unix/i686/oci817_8/libwebl ogicoci37.so`
		`lib/tru64unix/i686/oci817_8/libwebl ogicoxa37.so`

NSAPI Plug-ins	**For this plugin:**	**These shared libraries are supported:**
	iPlanet 3.6	`weblogic/lib/tru64unix/libproxy.so`
	iPlanet 4.0	`weblogic/lib/tru64unix/libproxy.so`
	iPlanet 4.1	`weblogic/lib/tru64unix/libproxy.so`
	Sun One 6.0 (These binaries are only available through customer support for WebLogic 5.1 Service Pack 1 users)	`weblogic/lib/tru64unix/libproxy.so`
Apache Plug-ins	**For these Apache versions:**	**This shared library is supported:**
	1.3.9 1.3.12 1.3.19	`weblogic/lib/tru64unix/mod_wl.so` (The `mod_wl.so` is a shared object for standard Apache (non EAPI). It is available beginning with WLS 5.1.0 Service Pack 7)

Known Issues

- Starting the Server with JDK 1.2.2

  JDK 1.2.2 is a supported JVM when using Weblogic 5.1 on a Tru64 Unix platform. However, the WebLogic Server's startup script needs to be modified in order for the server to start properly.

  The JIT(Just-In-Time compiler) must be turned off; whereby the `-Djava.compiler` directive needs to be passed to the JVM and the JIT disabled. For example:

```
/usr/opt/java122/bin/java -Djava.compiler= -ms128m -mx128m -classpath
./classes/boot
-Dweblogic.class.path=./license:./classes:./lib/weblogicaux.jar:./myser
ver/serverclasses
-Djava.security.manager -Djava.security.policy==weblogic.policy
weblogic.Server
```

- Port 80 Requirement

  As of WebLogic Service Pack 7, all 5.1 Service Packs will include a `libweblogicunix1.so` in order for HP Tru64 Unix 5.0 to be run on port 80.

CHAPTER 5

Hewlett-Packard HP/9000 with HP-UX 11.0 and 11i

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- For general information about HP-UX, see the Hewlett-Packard website.
- The HP-UX 11.0 default value for maximum threads per process is 64, which may be too low for the WebLogic Server and many Java server applications in general. When the value is too low, your WebLogic Server may have problems starting up.
- We recommend that you use the maximum kernel setting for the `max_thread_proc` and `maxusers` parameters. Setting `maxusers` to a higher value increases other machine-wide limits, such as `nkthread` (the maximum number of kernel threads). For example, we have reconfigured our kernel as follows:

Table 5-1 Recommended HP-UX 11.0 Kernel Settings

Parameter	Old Value	New Value	Description
`max_thread_proc`	64	1024	Maximum threads per process
`maxfiles`	60	256	Soft file limit per process
`maxusers`	32	256	Influences other parameters
`nkthread`	499	3635	Number of threads total on the system

Table 5-1 Recommended HP-UX 11.0 Kernel Settings

Parameter	Old Value	New Value	Description
`nproc`	276	2068	Maximum number of processes
`ncallout`	292	2084	Number of pending timeouts

Supported Releases

This section lists the releases supported for this platform. Unless otherwise noted, all service packs for each release are also supported.

- WebLogic 5.1

WebLogic 5.1

Operating System	HP-UX 11.0 and 11i
Required OS Patches	None
CPU	PA-RISC 100MHz or higher
RAM	64 MB minimum
Disk Space	64 MB

Supported Client JVM	• HP-UX SDK 1.3.1.06 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1.06-020625-14:20) Java HotSpot™ Server VM (build 1.3.1 1.3.1.06-JPSE_1.3.1.06_20020625 PA2.0, mixed mode) • HP-UX SDK 1.3.1_05 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1.05-020425-12:07) Java HotSpot™ Server VM (build 1.3.1 1.3.1.05-JPSE_1.3.1.05_20020425 PA2.0, mixed mode) • HP-UX SDK 1.3.1_02 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1.02-011206-02:17) Java HotSpot™ Server VM (build 1.3.1 1.3.1.02-JPSE_1.3.1.02_20011206 PA2.0, mixed mode) • HP-UX SDK 1.3.1_01 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1.01-release-010816-12:37) Java HotSpot™ Server VM (build 1.3.1 1.3.1.01-release-010816-13:34-PA _RISC2.0 PA2.0, mixed mode)	• WebLogic Server 5.1 SP13 is certified with JDK 1.3.1.06. It will support service packs 1.3.1.06 through 1.3.1.09. • WebLogic Server 5.1 SP12 is certified with JDK 1.3.1.05. It will support service packs 1.3.1.05 through 1.3.1.09. • WebLogic Server 5.1 SP11 is certified with JDK 1.3.1.02. It will support service packs 1.3.1.02 through 1.3.1.09. • WebLogic Server 5.1 SP10 is certified with JDK 1.3.1.01. It will support service packs 1.3.1.01 through 1.3.1.09.

Supported Client JVM, continued	• HP-UX SDK, for the Java™ 2 Platform Version 1.3, with Java™ 2 Runtime Environment Standard Edition and the HotSpot VM (build 1.0.1fcs jinteg:11/28/00-13:54 PA2.0, mixed mode) or the Classic VM (build jinteg:11/28/00-13:54, build jinteg:11/28/00-13:54, native threads, HP) **Note:** BEA recommends using this SDK with HP-UX 11.0. This SDK has been tested and, in addition, is undergoing some supplemental rigorous stress testing above and beyond our current standards. • SDK 1.3.0 • JDK 1.2.2_08 HotSpot • JDK 1.2.2_06 HotSpot VM (1.0.1fcs, mixed mode, PA2.0 build 1.2.2.06-00/10/23-PA_RISC2.0) • JDK 1.2.2_03 HotSpot • JDK 1.1.8_02

Supported Server JVM	• HP-UX SDK 1.3.1.06 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1.06-020625-14:20) Java HotSpot™ Server VM (build 1.3.1 1.3.1.06-JPSE_1.3.1.06_20020625 PA2.0, mixed mode) • HP-UX SDK 1.3.1_05 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1.05-020425-12:07) Java HotSpot™ Server VM (build 1.3.1 1.3.1.05-JPSE_1.3.1.05_20020425 PA2.0, mixed mode) • HP-UX SDK 1.3.1.02 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1.02-011206-02:17) Java HotSpot™ Server VM (build 1.3.1 1.3.1.02-JPSE_1.3.1.02_20011206 PA2.0, mixed mode) • HP-UX SDK 1.3.1_01 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1.01-release-010816-12:37) Java HotSpot™ Server VM (build 1.3.1 1.3.1.01-release-010816-13:34-PA _RISC2.0 PA2.0, mixed mode)	• WebLogic Server 5.1 SP13 is certified with JDK 1.3.1.06. It will support service packs 1.3.1.06 through 1.3.1.09. • WebLogic Server 5.1 SP12 is certified with JDK 1.3.1.05. It will support service packs 1.3.1.05 through 1.3.1.09. • WebLogic Server 5.1 SP11 is certified with JDK 1.3.1.02. It will support service packs 1.3.1.02 through 1.3.1.09. • WebLogic Server 5.1 SP10 is certified with JDK 1.3.1.01. It will support service packs 1.3.1.01 through 1.3.1.09.

Supported Server JVM, continued	• HP-UX SDK, for the Java™ 2 Platform Version 1.3, with Java™ 2 Runtime Environment Standard Edition and the HotSpot VM (build 1.0.1fcs jinteg:11/28/00-13:54 PA2.0, mixed mode) or the Classic VM (build jinteg:11/28/00-13:54, build jinteg:11/28/00-13:54, native threads, HP) **Note:** BEA recommends using this SDK with HP-UX 11.0. This SDK has been tested and, in addition, is undergoing some supplemental rigorous stress testing above and beyond our current standards. • SDK 1.3.0 • JDK 1.2.2_08 HotSpot • JDK 1.2.2_06 HotSpot VM (1.0.1fcs, mixed mode, PA2.0 build 1.2.2.06-00/10/23-PA_RISC2.0) • JDK 1.2.2_03 HotSpot • JDK 1.1.8_02	
Performance Pack	Performance Pack included.	
Node Manager	None.	

Oracle jDriver	**For this Oracle Server version:**
	Oracle Server 8.1.7
		These Oracle jDriver shared libraries are supported:
		`lib/hpux11/oci817_8/libweblogicoci37.sl`
		`lib/hpux11/oci817_8/libweblogicoxa37.sl`
		Note: Note: JDriver for Oracle 817 is not supported with JDK 1.3.1_02. BEA recommends using a type4 java driver for oracle connectivity with this VM.

NSAPI Plug-ins	**For this plugin:**	**This shared library is supported:**
	iPlanet 3.6	`weblogic/lib/hpux11/libproxy.sl`
	iPlanet 4.0 *Requires service pack 7 or higher for WLS 5.1*	`weblogic/lib/hpux11/libproxy.sl`
	iPlanet 4.1 *Requires service pack 7 or higher for WLS 5.1*	`weblogic/lib/hpux11/libproxy.sl`
	Sun One 6.0 *Requires service pack 11 or higher*	`weblogic/lib/hpux11/libproxy.sl`
Apache Plug-ins	**For this Apache version:**	**These shared libraries are supported:**
	HP Apache 1.3.x. See HP Apache-based Web Server v.1.3.x End of Support Announcement.	`weblogic/lib/hpux11/mod_wl.so` (Shared object for standard Apache - non EAPI installations) `weblogic/lib/hpux11/mod_wl_ssl.so` (Use for Apache + SSL/EAPI installations - Stronghold, modssl etc.) `weblogic/lib/hpux11/mod_wl_ssl_raven.so` (Use for Apache and FastStart installations)

CHAPTER 6

IBM AS/400e with OS/400 V4R4/V4R5

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- The AS/400 Developer Center is a good starting point for using WebLogic on the IBM AS/400. From the Developer Center, you can access documentation written specifically about using WebLogic software in an AS/400 environment, as well as documents relevant to all WebLogic users:

  WebLogic Server 5.1 Developer Center for AS/400e

Supported Releases

This section lists the releases supported for this platform:

- WebLogic 5.1 (Supported on OS/400 V4R4 and V4R5)

WebLogic 5.1

Operating System	OS/400 V4R4 and V4R5
Required OS Patches	None
CPU	Model 170e 266MHz or better
RAM	64MB RAM minimum
Disk Space	N/A
Supported Client JVM	JDK 1.1.6 JDK 1.2.2
Supported Server JVM	JDK 1.1.6 JDK 1.2.2
Performance Pack	None.
Node Manager	None.
Oracle jDriver	None.
NSAPI Plug-ins	None.
Apache Plug-ins	None.

Known Issues

- Due to a multicast socket problem, clustering does not work on this platform.
- Due to a JVM problem, compiling with WebLogic `rmic` with the `-idl` command does not work.
- Cloudscape databases do not work with the provided examples; you must use the DB2 that ships with OS/400.

CHAPTER 7

IBM Dynix/ptx

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- DYNIX/ptx is IBM's (formerly Sequent's) version of UNIX for NUMA-Q computers.

Supported Releases

This section lists the releases supported for this platform.

- WebLogic 5.1

WebLogic 5.1

Operating System	NUMA-Q Server
Required OS Patches	None.
CPU	Intel Pro
RAM	Tested with 1 GB
Disk Space	100 MB
Supported Client JVM	ptx 4.4.4/JSE V1.1.2
Supported Server JVM	ptx 4.4.4/JSE V1.1.2
Performance Pack	None.
Node Manager	None.
Oracle jDriver	None.
NSAPI Plug-ins	None.
Apache Plug-ins	None.

CHAPTER 8

IBM S/390 with OS/390

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- For general information about S/390, OS/390, and USS/390, see IBM's S/390 website.
- Graphical utilities, such as the WebLogic Console, must be executed on a client computer, since USS390 does not have a GUI environment.

Supported Releases

This section lists the releases supported for this platform.

- WebLogic 5.1

WebLogic 5.1

Operating System	OS/390 V2R8
Required OS Patches	None
CPU	IBM S/390 G5 or later models
RAM	128 MB Minimum

Disk Space	77 MB
Supported Client JVM	JDK 1.1.8
Supported Server JVM	JDK 1.1.8
Performance Pack	None
Node Manager	None
Oracle jDriver	None
NSAPI Plug-ins	None
Apache Plug-ins	None.

CHAPTER 9

Windows 2000 Server, Windows 2000 Advanced Server for IA-32

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.

Supported Releases

This section lists the releases supported for this platform.

- WebLogic 5.1

WebLogic 5.1

Operating System	Windows 2000 Server or Windows 2000 Advanced Server
Required OS Patches	None.
CPU	Pentium 200MHz or higher
RAM	64 MB minimum
Disk Space	80 MB (InstallShield) or 64MB (`.zip` archive)

Supported Client JVM	SunSoft SDK 1.3.1 with Java HotSpot™ Client VM	WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1.
Supported Server JVM	SunSoft SDK 1.3.1 with Java HotSpot™ Client VM	WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1.
Performance Pack	Included.	
Node Manager	None.	
Oracle jDriver	None.	
NSAPI Plug-ins	None.	
Apache Plug-ins	None.	

CHAPTER 10

Windows 2000 Professional for IA-32

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- This platform is supported for development only.

Supported Releases

This section lists the releases supported for this platform.

- WebLogic 5.1

WebLogic 5.1

Operating System	Windows 2000 Professional
Required OS Patches	None
CPU	Pentium 200MHz or higher
RAM	64 MB minimum
Disk Space	80 MB

Supported Client JVM	• SunSoft SDK 1.3.1 with Java HotSpot™ Client VM • SunSoft SDK 1.3.0 Java™ 2 Runtime Environment, Standard Edition (build 1.3.0) Classic VM (build 1.3.0, native threads, nojit), or Java™ 2 Runtime Environment, Standard Edition (build 1.3.0) Java HotSpot™ Client VM (build 1.3.0, mixed mode) • JDK 1.2.2 Classic VM (build JDK-1.2.2_007, native threads, symcjit) or, Classic VM (build JDK-1.2.2_006, native threads, symcjit) or, Classic VM (build JDK-1.2.2_005, native threads, symcjit) **Note:** BEA also supports "HotSpot VM (1.0.1, mixed mode, build g)", running on these JDK's. • JDK 1.1.7B	WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1.

Supported Server JVM	• SunSoft SDK 1.3.1 with Java HotSpot™ Client VM • SunSoft SDK 1.3.0 Java™ 2 Runtime Environment, Standard Edition (build 1.3.0) Classic VM (build 1.3.0, native threads, nojit), or Java™ 2 Runtime Environment, Standard Edition (build 1.3.0) Java HotSpot™ Client VM (build 1.3.0, mixed mode) • JDK 1.2.2 Classic VM (build JDK-1.2.2_007, native threads, symcjit) or, Classic VM (build JDK-1.2.2_006, native threads, symcjit) or, Classic VM (build JDK-1.2.2_005, native threads, symcjit) **Note:** BEA also supports "HotSpot VM (1.0.1, mixed mode, build g)", running on these JDK's. • JDK 1.1.7B	WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1.
Performance Pack	Included.	
Node Manager	None.	
Oracle jDriver	None.	
NSAPI Plug-ins	None.	
Apache Plug-ins	None.	

CHAPTER 11

Windows NT 4.0 for IA-32

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.

Supported Releases

This section lists the releases supported for this platform.

- WebLogic 5.1

WebLogic 5.1

Operating System	Windows NT 4.0
Required OS Patches	Service Pack 4 or higher
CPU	Pentium 200MHz or higher
RAM	64 MB minimum
Disk Space	80MB disk space (InstallShield) or 64MB (`.zip` archive)

Supported Client JVM	• SunSoft SDK 1.3.1 with Java HotSpot™ Client VM • SunSoft SDK 1.3.0 Java™ 2 Runtime Environment, Standard Edition (build 1.3.0) Classic VM (build 1.3.0, native threads, nojit), or Java™ 2 Runtime Environment, Standard Edition (build 1.3.0) Java HotSpot™ Client VM (build 1.3.0, mixed mode) • JDK 1.2.2 Classic VM (build JDK-1.2.2_007, native threads, symcjit) or, Classic VM (build JDK-1.2.2_006, native threads, symcjit) or, Classic VM (build JDK-1.2.2_005, native threads, symcjit) BEA also supports "HotSpot VM (1.0.1, mixed mode, build g)", running on these JDK's • Microsoft SDK for Java 4.0 • JDK 1.1.7B	**Client JVM Notes:** • WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1. • Due to Sun Bug IDs 4296365 and 4259286, JDK 1.2.2 with HotSpot is not supported for Windows NT on WebLogic Server.

Supported Server JVM	• SunSoft SDK 1.3.1 with Java HotSpot™ Client VM • SunSoft SDK 1.3.0 Java™ 2 Runtime Environment, Standard Edition (build 1.3.0) Classic VM (build 1.3.0, native threads, nojit), or Java™ 2 Runtime Environment, Standard Edition (build 1.3.0) Java HotSpot™ Client VM (build 1.3.0, mixed mode) • JDK 1.2.2 Classic VM (build JDK-1.2.2_007, native threads, symcjit) or, Classic VM (build JDK-1.2.2_006, native threads, symcjit) or, Classic VM (build JDK-1.2.2_005, native threads, symcjit) BEA also supports "HotSpot VM (1.0.1, mixed mode, build g)", running on these JDK's • Microsoft SDK for Java 4.0 • JDK 1.1.7B	**Server JVM Notes:** • WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1. • Due to Sun Bug IDs 4296365 and 4259286, JDK 1.2.2 with HotSpot is not supported for Windows NT on WebLogic Server.
Performance Pack	Included.	
Node Manager	None	
Oracle jDriver	**For this Oracle Server version:** Oracle 8.1.7	**This Oracle jDriver shared library is supported:** `bin/oci817_8/libweblogicoci37.dll`

NSAPI Plug-ins	**For this plugin:**	**This shared library is supported:**
	iPlanet 3.6	`weblogic/bin/proxy36.dll`
	iPlanet 4.0	`weblogic/bin/proxy36.dll`
	iPlanet 4.1.x	`weblogic/bin/proxy36.dll`
	Sun One 6.0 Requires 5.1 Service Pack 11 or higher	`weblogic/bin/proxy36.dll`
Apache Plug-ins	None	

CHAPTER 12

Red Hat Linux for IA-32

General Information

- WebLogic Server will end support for the Red Hat Linux 7.x starting December 30, 2003. For information, see Red Hat Linux 7.x End of Support Announcement.
- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- The WebLogic Server is tested on Linux with the following JDBC drivers:
  - Sybase jConnect for Sybase SQL Server
  - WebLogic jDriver for Microsoft SQL Server (formerly known as jdbcKona/MSSQLServer4)
  - WebLogic jDriver for Informix (formerly known as jdbcKona/Informix4)

Supported Releases

This section lists the releases supported for this platform.

- WebLogic 5.1

WebLogic 5.1

Operating System	Red Hat Linux 6.1, 6.2, 7.1 (Red Hat 7.1 with WebLogic 5.1 Service Pack 11 or higher)	
Required OS Patches	None.	
CPU	Intel IA-32	
RAM	64 MB minimum	
Disk Space	80 MB (InstallShield) or 64MB (`.zip` archive)	
Supported Client JVM	• JDK 1.3.1 (for Linux 7.1/WebLogic Service Pack 11 or higher required) with Java HotSpot™ Client VM • Sun Java 2 SDK v1.2.2 • IBM Linux 1.1.8	WebLogic 5.1 service pack releases SP11 and higher for Linux 7.1 were certified with SunSoft SDK 1.3.1 and will support any service pack of SunSoft SDK 1.3.1.
Supported Server JVM	• JDK 1.3.1 (for Linux 7.1/WebLogic Service Pack 11 or higher required) with Java HotSpot™ Client VM • Sun Java 2 SDK v1.2.2 • IBM Linux 1.1.8	WebLogic 5.1 service pack releases SP11 and higher for Linux 7.1 were certified with SunSoft SDK 1.3.1 and will support any service pack of SunSoft SDK 1.3.1.
Performance Pack	Included.	
Node Manager	None.	
Oracle jDriver	**For this Oracle Server version:** Oracle 8.1.7 (Service Pack 9 or higher)	

NSAPI Plug-ins	None.	

Apache Plug-ins	**For this Apache version:**
	1.3.x
		These shared libraries are supported:
		`weblogic/lib/linux/i686/mod_wl.so`
		(The `mod_wl.so` is a shared object for standard Apache (non EAPI).
		
		`weblogic/lib/linux/i686/mod_wl_ssl.so`
		(The file `mod_wl_ssl.so` is used for Apache + SSL/EAPI installations (Stronghold, modssl etc.).
		
		`weblogic/lib/linux/i686/mod_wl_ssl_raven.so`
		(For Apache+FastStart installations the `mod_wl_ssl_raven.so` is used. The `mod_wl_ssl_raven.so` file was required because FastStart applies frontpage patches which makes it incompatible with `mod_wl_ssl.so`.)

Known Issues

WebLogic Server does support the performance pack on RHAT Linux 6.1 for WebLogic Server 5.1 in conjunction with JDK 1.1.8 (Sun's 1.2.2 only supports green threads and is not supported with the performance pack).

Also, the Red Hat Linux `setuid` functionality doesn't work with the JDK 1.3.1_02 because of the current Linux thread model. If you use ports lower than 1024, you will not be able to use this function to switch from a root to a non-privileged user id.

CHAPTER 13

Sun Microsystems/Fujitsu SPARC with Solaris 2.5.1

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- WebLogic supports Sun Solaris on Sun Microsystems/Fujitsu SPARC computers. For information on Solaris, see Sun's Solaris web page.
- For Sun's support services info site, see http://www.sun.com/service/support. If you have a support contract for the use of Solaris platforms, you may be able to obtain support for general Java issues (for example, JVM problems).
- You can run WebLogic software on Solaris using the SDK available from JavaSoft. BEA recommends using the latest BEA supported version of the JVM's available, except where noted on our Supported Configurations document.
- The JIT (Just-In-Time) Compiler is only supported for 1.2.2_05a while using the `export _JVM_ARGS="inline_instrs_jit=0"` directive.This is due to Sun's bugs: #4333396 (JVM crash during JIT) and #4333696. The error message reads: `SIGABRT and JIT coredump: panic: Unexpected operand kind(10) in j86OpndImage`. These also affect 1.2.1_04, so that you must Disable aggressive inlining by the jit with: `export _JIT_ARGS=novinline.`
- For information on suggested values for some Solaris TCP and VM-related tunable parameters, see Solaris Tunable Parameters.

Supported Releases

This section lists the releases supported for this platform. Unless otherwise noted, all service packs for each release are also supported.

- WebLogic 5.1

WebLogic 5.1

Operating System	Solaris 2.5.1
Required OS Patches	None
CPU	UltraSPARC 168MHz or better
RAM	64 MB minimum
Disk Space	64 MB
Supported Client JVM	• JDK 1.2.2_06 See BEA recommendations at JDK 1.2.2_06. • JDK 1.2.2_05a See BEA recommendations at JDK 1.2.2_05a. • SunSoft JDK 1.2.1_04 See BEA recommendations at SunSoft JDK 1.2.1_04. • SunSoft JDK 1.1.8_12 • SunSoft JDK 1.1.7_08a See BEA recommendations at JDK 1.1.7_08a.

Supported Server JVM	• JDK 1.2.2_06 See BEA recommendations at JDK 1.2.2_06. • JDK 1.2.2_05a See BEA recommendations at JDK 1.2.2_05a. • SunSoft JDK 1.2.1_04 See BEA recommendations at SunSoft JDK 1.2.1_04. • SunSoft JDK 1.1.8_12 • SunSoft JDK 1.1.7_08a See BEA recommendations at JDK 1.1.7_08a.	
Performance Pack	Included.	
Node Manager	None.	

Oracle jDriver	**For this Oracle Server version:**
	Oracle 8.1.7
	Requires Service Pack 10 or higher
		These Oracle jDriver shared libraries are supported:
		`lib/solaris/oci817_8/libweblogicoci37.so`
		`lib/solaris/oci817_8/libweblogicoxa37.so`

NSAPI Plug-ins	**For this plugin:**	**This shared library is supported:**
	iPlanet 3.6	`weblogic/lib/solaris/libproxy.so`
	iPlanet 4.0	`weblogic/lib/solaris/libproxy.so`
	iPlanet 4.1.x	`weblogic/lib/solaris/libproxy.so`
	Sun One 6.0 WebLogic 5.1 Service Pack 11 and later	`weblogic/lib/solaris/libproxy.so`
Apache Plug-ins	**For this Apache version:**	**These shared libraries are supported:**

	1.3.x	`weblogic/lib/solaris/mod_wl.so`
		The `mod_wl.so` is a shared object for standard Apache (non EAPI).
		
		`weblogic/lib/solaris/mod_wl_ssl.so`
		The file `mod_wl_ssl.so` is used for Apache + SSL/EAPI installations (Stronghold, modssl etc). It is available beginning with WLS 5.1.0 Service Pack 4.
		
		`weblogic/lib/solaris/mod_wl_ssl_raven.so`
		For Apache+FastStart installations the `mod_wl_ssl_raven.so` is used.
		The `mod_wl_ssl_raven.so` file was required because FastStart applies frontpage patches which makes it incompatible with `mod_wl_ssl.so`. It is available beginning with WLS 5.1.0 Service Pack 4)

Known Issues

Problems with Solaris IP numbers

We have encountered a problem where some Solaris JVMs report an incorrect IP number. The reported IP number for the machine is the first number in the `/etc/hosts` file, which, unfortunately, is frequently `127.0.0.1`, the standard loopback network address for the machine.

This problem affects licenses, where a license is granted for a specific IP number. We have also observed this problem in a cluster where an HTTP request is passed by a web server or another WebLogic Server via the HttpClusterServlet, to a WebLogic Server running on a Solaris machine with this configuration problem. The `127.0.0.1` IP number is carried with the server's response. Subsequent calls to that machine time out because of the incorrect IP number.

You can run the command `java utils.netAddresses` to see if you have this problem on your Solaris computer. If you see `127.0.0.1` in the output, ask the system administrator to move the computer's network IP address to the beginning of the `/etc/hosts` file.

JDK 1.2.2_06

BEA recommends using the 1.2.2_05a version with the directive `export _JVM_ARGS="inline_instrs_jit=0".` This will disable aggressive inlining by the JIT while allowing it to perform other performance enhancing tasks. The 1.2.2_05a version has been supported by BEA. There is a known garbage collection related deadlock condition, bug #4351991, with JDK 1.2.2_06. If you wish to use the 1.2.2_06 version, emergency relief that fixes bug #4351991 will be available through Sun, up until the release of 1.2.2_07 or the full BEA qualification of a later Sun JDK. Please contact your Sun support representative with any questions regarding this relief.

JDK 1.2.2_05a

BEA recommends using the 1.2.2_05a version with the directive `export _JVM_ARGS="inline_instrs_jit=0".` This disables aggressive inlining by the JIT while allowing it to perform other performance enhancing tasks.

JavaSoft JDK 1.2.2

The wrapper script for the Solaris JDK 1.2.2 ignores the value of `$THREADS_FLAG` and defaults to using green threads unless you supply the `-native` option. WebLogic Server is not supported with green threads. Please specify the `-native` option on all of your `java` commands.

SunSoft JDK 1.2.1_04

You must disable aggressive inlining by the JIT with: `export _JIT_ARGS=novinline`

JDK 1.1.7_08a

Bug ID 4134584 in Sun's Bug Parade, describes a bug in JDK 1.1.7 on Solaris. This bug will occasionally cause a server crash with the error `"*** panic: 16-bit string hash table overflow"`. Because of this, we recommend that you use JDK 1.2 on Solaris.)

WebLogic Clusters and Solaris 2.5.1

WebLogic Server clusters require Solaris 2.6. Clusters are not supported on Solaris 2.5.1 because the multicast implementation prevents the clustered servers from communicating with each other. Solaris 2.5.1 should not be used in a WebLogic Server cluster.

Tuning Solaris File Descriptor Limits

On Solaris, each user account has a certain limited number of file descriptors. Use the `ulimit` command to print or set resource limits. A resource limit is a pair of values that specify the current (soft) limit and the maximum (hard) limit. You can modify the hard limit in `/etc/system`. You must reboot your machine anytime you modify `/etc/system`.

Note: Do not change the default soft limit. It has the potential to affect many processes on the server and will not affect WebLogic Server.

You must have adequate permissions to use the `ulimit` command. Any user may lower a hard limit. Only a super-user may raise a hard limit.

Viewing and Adjusting File Descriptor Limits

To view and adjust file descriptor limits:

1. Use the `ulimit` command to print current resource limits.
   ```
   ulimit
   ```
2. Set the hard limit value in /etc/system, according to your needs. For example:
   ```
   set rlim_fd_max=4096 /* hard limit */
   ```
3. Restart WebLogic Server. A message similar to the following appears in the startup log:
   ```
   <Jul 30, 2001 9:16:11 AM EDT> <Info> <Posix Performance Pack> <System
   has file descriptor limits of - soft: '256', hard: '4096'>
   <Jul 30, 2001 9:16:11 AM EDT> <Info> <Posix Performance Pack> <Using
   effective file descriptor limit of: '4096' open sockets/files.>
   ```

Solaris Tunable Parameters

Warning: Do not modify these parameters before first consulting your Solaris Administrator.

This section presents suggested values for some Solaris TCP and VM-related tunable parameters.

For more information about Solaris configuration, see the Solaris FAQ: http://www.science.uva.nl/pub/solaris/solaris2/index.html.

Setting TCP Parameters with the ndd Command

Set the following TCP tuning parameters with the `ndd` command. Use the following syntax:

```
ndd -set driver parameter
```

For example:

```
ndd -set /dev/ip ip_ignore_redirect 1
```

Table 13-1 Suggested TCP Parameter Values

Parameter	Suggested Value
/dev/ip ip_ignore_redirect	1
/dev/tcp tcp_conn_grace_period	500
/dev/tcp tcp_conn_req_max_q	8096 (*)
/dev/tcp tcp_conn_req_max_q0	8096 (*)
/dev/tcp tcp_conn_req_min	1
/dev/tcp tcp_cwnd_max	65534
/dev/tcp tcp_fin_wait_2_flush_interval	16000
/dev/tcp tcp_ip_abort_cinterval	60000
/dev/tcp tcp_ip_abort_interval	60000
/dev/tcp tcp_keepalive_interval	90000
/dev/tcp tcp_recv_hiwat	32768
/dev/tcp tcp_rexmit_interval_initial	3000

Table 13-1 Suggested TCP Parameter Values

Parameter	Suggested Value
/dev/tcp tcp_rexmit_interval_max	10000
/dev/tcp tcp_rexmit_interval_min	3000
/dev/tcp tcp_slow_start_initial	2
/dev/tcp tcp_time_wait_interval	60000 (**)
/dev/tcp tcp_xmit_hiwat	32768

Set parameters followed with an asterisk based on the number of concurrent connections against the web server.

For Solaris 2.5.1 and 2.6, `tcp_time_wait_interval` has changed to:

```
ndd -set /dev/tcp tcp_close_wait_interval 60000
```

The `tcp_close_wait_interval` parameter determines the time interval that a TCP socket is kept alive after issuing a close call. The default value of this parameter on Solaris is four minutes. When a large number of clients connect for a short amount of time, holding these socket resources can have a significant negative impact on performance. Setting this parameter to a value of 60000 (60 seconds) has shown a significant throughput enhancement when running benchmark JSP tests on Solaris.

You might want to reduce this setting further if the server gets backed up with a queue of half-opened connections.

Viewing TCP Parameters with the netstat -sP tcp Command

Use `netstat -sP tcp` to view the following parameters:

Table 13-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpActiveOpens	413995
tcpAttemptFails	15937

Table 13-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpCurrEstab	13
tcpEstabResets	621691
tcpHalfOpenDrop	0
tcpInAckBytes	4290207863
tcpInAckSegs	43347796
tcpInAckUnsent	0
tcpInClosed	1700
tcpInDupAck	1230270
tcpInDupBytes	12551831
tcpInDupSegs	7023
tcpInInorderBytes	3228433225
tcpInInorderSegs	42175934
tcpInPartDupBytes	4690354
tcpInPartDupSegs	7023
tcpInPastWinBytes	486243985
tcpInPastWinSegs	178
tcpInSegs	64462930
tcpInUnorderBytes	258926562
tcpInUnorderSegs	206376
tcpInWinProbe	7584
tcpInWinUpdate	2754

Table 13-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpListenDrop	123865 At run time, if `tcpListenDrop` is non-zero and increasing, use the `ndd` command to increase `tcp_conn_req_max_q`.
tcpListenDropQ0	0 If `tcpListenDropQ0` is non-zero and increasing, use the `ndd` command to increase `tcp_conn_req_max_q0`.
tcpMaxConn	-1
tcpOutAck	12416786
tcpOutAckDelayed	3223013
tcpOutControl	4296229
tcpOutDataBytes	4290849539
tcpOutDataSegs	55577630
tcpOutFastRetrans	42726
tcpOutRsts	2735468
tcpOutSackRetrans	223
tcpOutSegs	68076290
tcpOutUrg	221
tcpOutWinProbe	4193
tcpOutWinUpdate	13576
tcpPassiveOpens	655052
tcpRetransBytes	148287157
tcpRetransSegs	242760

Table 13-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpRtoAlgorithm	4
tcpRtoMax	60000
tcpRtoMin	200
tcpRttNoUpdate	95781
tcpRttUpdate	42878526
tcpTimKeepalive	12798
tcpTimKeepaliveDrop	13
tcpTimKeepaliveProbe	2408
tcpTimRetrans	150215
tcpTimRetransDrop	526

Setting tcp_conn_hash_size Parameter in /etc/system

You must reboot your machine anytime you modify `/etc/system` parameters. Set the value of the `tcp_conn_hash_size` parameter in the `/etc/system` file:

```
set tcp:tcp_conn_hash_size=8192
```

VM Parameters

On JDK 1.2.2_05a, set the following parameter in the java command line:

```
java <other command line options>
-Xgenconfig:64m,64m,semispaces:64m,512m,markcompact weblogic.Server
```

See also "Tuning Java Virtual Machines" in the BEA WebLogic Server Performance and Tuning guide.

CHAPTER 14

Sun Microsystems/Fujitsu SPARC with Solaris 2.6

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- WebLogic supports Sun Solaris on Sun Microsystems/Fujitsu SPARC computers. For information on Solaris, see Sun's Solaris web page.
- For Sun's support services info site, see http://www.sun.com/service/support. If you have a support contract for the use of Solaris platforms, you may be able to obtain support for general Java issues (for example, JVM problems).
- You can run WebLogic software on Solaris using the SDK available from JavaSoft. BEA recommends using the latest BEA supported version of the JVM's available, except where noted on our Supported Configurations document.
- The JIT (Just-In-Time) Compiler is only supported for 1.2.2_05a while using the `export _JVM_ARGS="inline_instrs_jit=0"` directive.This is due to Sun's bugs: #4333396 (JVM crash during JIT) and #4333696. The error message reads: `SIGABRT and JIT coredump: panic: Unexpected operand kind(10) in j86OpndImage`. These also affect 1.2.1_04, so that you must Disable aggressive inlining by the jit with: `export _JIT_ARGS=novinline.`
- 7.3.4 jDriver for Oracle is not supported with JDK 1.3.1.
- WebLogic Server clusters require Solaris 2.6. Clusters are not supported on Solaris 2.5.1 because the multicast implementation prevents the clustered servers from communicating with each other. Solaris 2.5.1 should not be used in a WebLogic Server cluster.

- We have encountered a problem where some Solaris JVMs report an incorrect IP number. The reported IP number for the machine is the first number in the `/etc/hosts` file, which, unfortunately, is frequently `127.0.0.1`, the standard loopback network address for the machine. This problem affects licenses, where a license is granted for a specific IP number. We have also observed this problem in a cluster where an HTTP request is passed by a web server or another WebLogic Server via the HttpClusterServlet, to a WebLogic Server running on a Solaris machine with this configuration problem. The `127.0.0.1` IP number is carried with the server's response. Subsequent calls to that machine time out because of the incorrect IP number. You can run the command `java utils.netAddresses` to see if you have this problem on your Solaris computer. If you see `127.0.0.1` in the output, ask the system administrator to move the computer's network IP address to the beginning of the `/etc/hosts` file.
- For information on suggested values for some Solaris TCP and VM-related tunable parameters, see Solaris Tunable Parameters.

Supported Releases

This section lists the releases supported for this platform. Unless otherwise noted, all service packs for each release are also supported.

- WebLogic 5.1

Known JDK Issues

Problems with JDK 1.3 crashing

If you have problems with OutOfMemory errors and the JVM crashing with JDK 1.3, try setting: -XX:MaxPermSize=128m. There is currently an open bug on Sun's bug parade that describes this problem. See http://developer.java.sun.com/developer/bugParade/bugs/4390238.html.

JDK 1.2.2_06

BEA recommends using the 1.2.2_05a version with the directive `export _JVM_ARGS="inline_instrs_jit=0".` This will disable aggressive inlining by the JIT while allowing it to perform other performance enhancing tasks. The 1.2.2_05a version has been supported by BEA. There is a known garbage collection related deadlock condition, bug #4351991, with JDK 1.2.2_06. If you wish to use the 1.2.2_06 version, emergency relief that fixes bug #4351991 will be available through Sun, up until the release of 1.2.2_07 or the full BEA

qualification of a later Sun JDK. Please contact your Sun support representative with any questions regarding this relief.

JDK 1.2.2_05a

BEA recommends using the 1.2.2_05a version with the directive `export _JVM_ARGS="inline_instrs_jit=0"`. This disables aggressive inlining by the JIT while allowing it to perform other performance enhancing tasks.

JDK 1.2.2

The wrapper script for the Solaris JDK 1.2.2 ignores the value of `$THREADS_FLAG` and defaults to using green threads unless you supply the `-native` option. WebLogic Server is not supported with green threads. Please specify the `-native` option on all of your `java` commands.

JDK 1.2.1_04

You must disable aggressive inlining by the JIT with: `export _JIT_ARGS=novinline`

JDK 1.1.7_08a

Bug ID 4134584 in Sun's Bug Parade, describes a bug in JDK 1.1.7 on Solaris. This bug will occasionally cause a server crash with the error `"*** panic: 16-bit string hash table overflow"`. Because of this, we recommend that you use JDK 1.2 on Solaris.)

WebLogic 5.1

Operating System	Solaris 2.6
Required OS Patches	None
CPU	UltraSPARC 168MHz or better
RAM	64 MB minimum
Disk Space	64 MB

Supported Client JVM	• SunSoft SDK 1.3.1 with Java HotSpot™ Client VM • SunSoft SDK 1.3.0 Java™ 2 SDK Runtime Environment, Standard Edition (build 1.3.0), with Java HotSpot™ Client VM (build 1.3.0, mixed mode) • JDK 1.2.2_09 (build Solaris_JDK_1.2.2_09, native threads, sunwjit) • JDK 1.2.2_07 (build Solaris_JDK_1.2.2_07, native threads, sunwjit) • JDK 1.2.2_06 See Bea recommendations at JDK 1.2.2_06. • JDK 1.2.2_05a See Bea recommendations at JDK 1.2.2_05a. • SunSoft JDK 1.2.1_04 See BEA recommendations at JDK 1.2.1_04. • SunSoft JDK 1.1.8_12 • SunSoft JDK 1.1.7_08a See Bea recommendations at JDK 1.1.7_08a.	WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1.

Supported Server JVM	• SunSoft SDK 1.3.1 with Java HotSpot™ Client VM • SunSoft SDK 1.3.0 Java™ 2 SDK Runtime Environment, Standard Edition (build 1.3.0), with Java HotSpot™ Client VM (build 1.3.0, mixed mode) • JDK 1.2.2_09 (build Solaris_JDK_1.2.2_09, native threads, sunwjit) • JDK 1.2.2_07 (build Solaris_JDK_1.2.2_07, native threads, sunwjit) • JDK 1.2.2_06 See Bea recommendations at JDK 1.2.2_06. • JDK 1.2.2_05a See Bea recommendations at JDK 1.2.2_05a. • SunSoft JDK 1.1.8_12 • SunSoft JDK 1.1.7_08a See Bea recommendations at JDK 1.1.7_08a.	WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1.
Performance Pack	Included.	
Node Manager	None.	

Oracle jDriver	**For this Oracle Server version:**
	Oracle 8.1.7
	Requires Service Pack 10 or higher
		These Oracle jDriver shared libraries are supported:
		`lib/solaris/oci817_8/libweblogicoci37.so`
		`lib/solaris/oci817_8/libweblogicoxa37.so`

NSAPI Plug-ins	**For this plugin:**	**This shared library is supported:**
	iPlanet 3.6	`weblogic/lib/solaris/libproxy.so`
	iPlanet 4.0	`weblogic/lib/solaris/libproxy.so`
	iPlanet 4.1.x	`weblogic/lib/solaris/libproxy.so`
	Sun One 6.0 WebLogic 5.1 Service Pack 11 and later	`weblogic/lib/solaris/libproxy.so`
Apache Plug-ins	**For this Apache version:**	**These shared libraries are supported:**

	1.3.x	`weblogic/lib/solaris/mod_wl.so`
		The `mod_wl.so` is a shared object for standard Apache (non EAPI).
		
		`weblogic/lib/solaris/mod_wl_ssl.so`
		The file `mod_wl_ssl.so` is used for Apache + SSL/EAPI installations (Stronghold, modssl etc). It is available beginning with WLS 5.1.0 Service Pack 4.
		
		`weblogic/lib/solaris/mod_wl_ssl_raven.so`
		For Apache+FastStart installations the `mod_wl_ssl_raven.so` is used.
		The `mod_wl_ssl_raven.so` file was required because FastStart applies frontpage patches which makes it incompatible with `mod_wl_ssl.so`. It is available beginning with WLS 5.1.0 Service Pack 4)

Solaris Tunable Parameters

Warning: Do not modify these parameters before first consulting your Solaris Administrator.

This section presents suggested values for some Solaris TCP and VM-related tunable parameters.

For more information about Solaris configuration, see the Solaris FAQ: http://www.science.uva.nl/pub/solaris/solaris2/index.html.

Tuning Solaris File Descriptor Limits

On Solaris, each user account has a certain limited number of file descriptors. Use the `ulimit` command to print or set resource limits. A resource limit is a pair of values that specify the current (soft) limit and the maximum (hard) limit. You can modify the hard limit in `/etc/system`. You must reboot your machine anytime you modify `/etc/system`.

Note: Do not change the default soft limit. It has the potential to affect many processes on the server and will not affect WebLogic Server.

You must have adequate permissions to use the `ulimit` command. Any user may lower a hard limit. Only a super-user may raise a hard limit.

Viewing and Adjusting File Descriptor Limits

To view and adjust file descriptor limits:

1. Use the `ulimit` command to print current resource limits.
   ```
   ulimit
   ```
2. Set the hard limit value in /etc/system, according to your needs. For example:
   ```
   set rlim_fd_max=4096 /* hard limit */
   ```
3. Restart WebLogic Server. A message similar to the following appears in the startup log:
   ```
   <Jul 30, 2001 9:16:11 AM EDT> <Info> <Posix Performance Pack> <System
   has file descriptor limits of - soft: '256', hard: '4096'>
   <Jul 30, 2001 9:16:11 AM EDT> <Info> <Posix Performance Pack> <Using
   effective file descriptor limit of: '4096' open sockets/files.>
   ```

Setting TCP Parameters with the ndd Command

Set the following TCP tuning parameters with the `ndd` command. Use the following syntax:

```
ndd -set driver parameter
```

For example:

```
ndd -set /dev/ip ip_ignore_redirect 1
```

Table 14-1 Suggested TCP Parameter Values

Parameter	Suggested Value
/dev/ip ip_ignore_redirect	1
/dev/tcp tcp_conn_grace_period	500
/dev/tcp tcp_conn_req_max_q	8096 (*)
/dev/tcp tcp_conn_req_max_q0	8096 (*)
/dev/tcp tcp_conn_req_min	1
/dev/tcp tcp_cwnd_max	65534
/dev/tcp tcp_fin_wait_2_flush_interval	16000
/dev/tcp tcp_ip_abort_cinterval	60000
/dev/tcp tcp_ip_abort_interval	60000
/dev/tcp tcp_keepalive_interval	90000
/dev/tcp tcp_recv_hiwat	32768
/dev/tcp tcp_rexmit_interval_initial	3000
/dev/tcp tcp_rexmit_interval_max	10000
/dev/tcp tcp_rexmit_interval_min	3000
/dev/tcp tcp_slow_start_initial	2
/dev/tcp tcp_time_wait_interval	60000 (**)
/dev/tcp tcp_xmit_hiwat	32768

Set parameters followed with an asterisk based on the number of concurrent connections against the web server.

For Solaris 2.6, `tcp_time_wait_interval` has changed to:

```
ndd -set /dev/tcp tcp_close_wait_interval 60000
```

The `tcp_close_wait_interval` parameter determines the time interval that a TCP socket is kept alive after issuing a close call. The default value of this parameter on Solaris is four minutes. When a large number of clients connect for a short amount of time, holding these socket resources can have a significant negative impact on performance. Setting this parameter to a value of 60000 (60 seconds) has shown a significant throughput enhancement when running benchmark JSP tests on Solaris.

You might want to reduce this setting further if the server gets backed up with a queue of half-opened connections.

Viewing TCP Parameters with the netstat -sP tcp Command

Use `netstat -sP tcp` to view the following parameters:

Table 14-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpActiveOpens	413995
tcpAttemptFails	15937
tcpCurrEstab	13
tcpEstabResets	621691
tcpHalfOpenDrop	0
tcpInAckBytes	4290207863
tcpInAckSegs	43347796
tcpInAckUnsent	0
tcpInClosed	1700
tcpInDupAck	1230270
tcpInDupBytes	12551831
tcpInDupSegs	7023
tcpInInorderBytes	3228433225

Table 14-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpInInorderSegs	42175934
tcpInPartDupBytes	4690354
tcpInPartDupSegs	7023
tcpInPastWinBytes	486243985
tcpInPastWinSegs	178
tcpInSegs	64462930
tcpInUnorderBytes	258926562
tcpInUnorderSegs	206376
tcpInWinProbe	7584
tcpInWinUpdate	2754
tcpListenDrop	123865 At run time, if `tcpListenDrop` is non-zero and increasing, use the `ndd` command to increase `tcp_conn_req_max_q`.
tcpListenDropQ0	0 If `tcpListenDropQ0` is non-zero and increasing, use the `ndd` command to increase `tcp_conn_req_max_q0`.
tcpMaxConn	-1
tcpOutAck	12416786
tcpOutAckDelayed	3223013
tcpOutControl	4296229
tcpOutDataBytes	4290849539
tcpOutDataSegs	55577630

Table 14-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpOutFastRetrans	42726
tcpOutRsts	2735468
tcpOutSackRetrans	223
tcpOutSegs	68076290
tcpOutUrg	221
tcpOutWinProbe	4193
tcpOutWinUpdate	13576
tcpPassiveOpens	655052
tcpRetransBytes	148287157
tcpRetransSegs	242760
tcpRtoAlgorithm	4
tcpRtoMax	60000
tcpRtoMin	200
tcpRttNoUpdate	95781
tcpRttUpdate	42878526
tcpTimKeepalive	12798
tcpTimKeepaliveDrop	13
tcpTimKeepaliveProbe	2408
tcpTimRetrans	150215
tcpTimRetransDrop	526

Setting tcp_conn_hash_size Parameter in /etc/system

You must reboot your machine anytime you modify `/etc/system` parameters. Set the value of the `tcp_conn_hash_size` parameter in the `/etc/system` file:

```
set tcp:tcp_conn_hash_size=8192
```

VM Parameters

On JDK 1.2.2_05a, set the following parameter in the java command line:

```
java <other command line options>
-Xgenconfig:64m,64m,semispaces:64m,512m,markcompact weblogic.Server
```

See also "Tuning Java Virtual Machines" in the BEA WebLogic Server Performance and Tuning guide:

For a detailed explanation of the JDK 1.3 VM, see:

http://java.sun.com/j2se/docs/PerformanceFAQ.html

http://java.sun.com/j2se/docs/VMOptions.html

CHAPTER 15

Sun Microsystems/Fujitsu SPARC with Solaris 2.7

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- WebLogic supports Sun Solaris on Sun Microsystems/Fujitsu SPARC computers. For information on Solaris, see Sun's Solaris web page.
- For Sun's support services info site, see http://www.sun.com/service/support. If you have a support contract for the use of Solaris platforms, you may be able to obtain support for general Java issues (for example, JVM problems).
- You can run WebLogic software on Solaris using the SDK available from JavaSoft. BEA recommends using the latest BEA supported version of the JVM's available, except where noted on our Supported Configurations document.
- The JIT (Just-In-Time) Compiler is only supported for 1.2.2_05a while using the `export _JVM_ARGS="inline_instrs_jit=0"` directive.This is due to Sun's bugs: #4333396 (JVM crash during JIT) and #4333696. The error message reads: `SIGABRT and JIT coredump: panic: Unexpected operand kind(10) in j86OpndImage`. These also affect 1.2.1_04, so that you must Disable aggressive inlining by the jit with: `export _JIT_ARGS=novinline.`
- 7.3.4 jDriver for Oracle is not supported with JDK 1.3.1.
- We have encountered a problem where some Solaris JVMs report an incorrect IP number. The reported IP number for the machine is the first number in the `/etc/hosts` file, which, unfortunately, is frequently `127.0.0.1`, the standard loopback network address for the

machine. This problem affects licenses, where a license is granted for a specific IP number. We have also observed this problem in a cluster where an HTTP request is passed by a web server or another WebLogic Server via the HttpClusterServlet, to a WebLogic Server running on a Solaris machine with this configuration problem. The `127.0.0.1` IP number is carried with the server's response. Subsequent calls to that machine time out because of the incorrect IP number. You can run the command `java utils.netAddresses` to see if you have this problem on your Solaris computer. If you see `127.0.0.1` in the output, ask the system administrator to move the computer's network IP address to the beginning of the `/etc/hosts` file.

- For information on suggested values for some Solaris TCP and VM-related tunable parameters, see Solaris Tunable Parameters

Supported Releases

This section lists the releases supported for this platform. Unless otherwise noted, all service packs for each release are also supported.

- WebLogic 5.1

Known JDK Issues

Problems with JDK 1.3 crashing

If you have problems with OutOfMemory errors and the JVM crashing with JDK 1.3, try setting: -XX:MaxPermSize=128m. There is currently an open bug on Sun's bug parade that describes this problem. See http://developer.java.sun.com/developer/bugParade/bugs/4390238.html.

JDK 1.2.2_06

BEA recommends using the 1.2.2_05a version with the directive `export _JVM_ARGS="inline_instrs_jit=0".` This will disable aggressive inlining by the JIT while allowing it to perform other performance enhancing tasks. The 1.2.2_05a version has been supported by BEA. There is a known garbage collection related deadlock condition, bug #4351991, with JDK 1.2.2_06. If you wish to use the 1.2.2_06 version, emergency relief that fixes bug #4351991 will be available through Sun, up until the release of 1.2.2_07 or the full BEA qualification of a later Sun JDK. Please contact your Sun support representative with any questions regarding this relief.

JDK 1.2.2_05a

BEA recommends using the 1.2.2_05a version with the directive `export _JVM_ARGS="inline_instrs_jit=0"`. This disables aggressive inlining by the JIT while allowing it to perform other performance enhancing tasks.

JDK 1.2.2

The wrapper script for the Solaris JDK 1.2.2 ignores the value of `$THREADS_FLAG` and defaults to using green threads unless you supply the `-native` option. WebLogic Server is not supported with green threads. Please specify the `-native` option on all of your `java` commands.

JDK 1.2.1_04

You must disable aggressive inlining by the JIT with: `export _JIT_ARGS=novinline`

JDK 1.1.7_08a

Bug ID 4134584 in Sun's Bug Parade, describes a bug in JDK 1.1.7 on Solaris. This bug will occasionally cause a server crash with the error `"*** panic: 16-bit string hash table overflow"`. Because of this, we recommend that you use JDK 1.2 on Solaris.)

WebLogic 5.1

Operating System	Solaris 2.7
Required OS Patches	None
CPU	UltraSPARC 168MHz or better
RAM	64 MB minimum
Disk Space	64 MB

Supported Client JVM	• SunSoft SDK 1.3.1_06 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_06-b01) Java HotSpot™ Client VM (build 1.3.1_06-b01, mixed mode) SunSoft SDK 1.3.1 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1-b24) with Java HotSpot™ Client VM (build 1.3.1-b24, mixed mode) • SunSoft SDK 1.3.0 Java™ 2 SDK Runtime Environment, Standard Edition (build 1.3.0), with Java HotSpot™ Client VM (build 1.3.0, mixed mode) • JDK 1.2.2_09 (build Solaris_JDK_1.2.2_09, native threads, sunwjit) • JDK 1.2.2_07 (build Solaris_JDK_1.2.2_07, native threads, sunwjit) • JDK 1.2.2_06 See Bea recommendations at JDK 1.2.2_06. • JDK 1.2.2_05a See Bea recommendations at JDK 1.2.2_05a. • SunSoft JDK 1.2.1_04 See BEA recommendations at JDK 1.2.1_04. • SunSoft JDK 1.1.8_12 • SunSoft JDK 1.1.7_08a See Bea recommendations at JDK 1.1.7_08a.	WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1.

Supported Server JVM	• SunSoft SDK 1.3.1_06 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_06-b01) Java HotSpot™ Client VM (build 1.3.1_06-b01, mixed mode) SunSoft SDK 1.3.1_02 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_02-b02) with Java HotSpot™ Client VM (build 1.3.1_02-b02, mixed mode) • SunSoft SDK 1.3.1 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1-b24) with Java HotSpot™ Client VM (build 1.3.1-b24, mixed mode) • SunSoft SDK 1.3.0 Java™ 2 SDK Runtime Environment, Standard Edition (build 1.3.0), with Java HotSpot™ Client VM (build 1.3.0, mixed mode) • JDK 1.2.2_09 (build Solaris_JDK_1.2.2_09, native threads, sunwjit) • JDK 1.2.2_07 (build Solaris_JDK_1.2.2_07, native threads, sunwjit) • JDK 1.2.2_06 See Bea recommendations at JDK 1.2.2_06. • JDK 1.2.2_05a See Bea recommendations at JDK 1.2.2_05a. • SunSoft JDK 1.1.8_12 • SunSoft JDK 1.1.7_08a See Bea recommendations at JDK 1.1.7_08a.	WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1.
Performance Pack	Included.	

Node Manager	None.	

Oracle jDriver	**For this Oracle Server version:**
	Oracle 8.1.7
	Requires Service Pack 10 or higher
		These Oracle jDriver shared libraries are supported:
		`lib/solaris/oci817_8/libweblogicoci37.so`
		`lib/solaris/oci817_8/libweblogicoxa37.so`

NSAPI Plug-ins	**For this plugin:** iPlanet 3.6	**This shared library is supported:** `lib/solaris/libproxy.so`
	iPlanet 4.0	`lib/solaris/libproxy.so`
	iPlanet 4.1.x	`lib/solaris/libproxy.so`
	Sun One 6.0 WebLogic 5.1 Service Pack 11 and later	`lib/solaris/libproxy.so`

Apache Plug-ins	**For this Apache version:**
	1.3.x
		These shared libraries are supported:
		`lib/solaris/mod_wl.so`
		The `mod_wl.so` is a shared object for standard Apache (non EAPI).
		
		`lib/solaris/mod_wl_ssl.so`
		The file `mod_wl_ssl.so` is used for Apache + SSL/EAPI installations (Stronghold, modssl etc). It is available beginning with WLS 5.1.0 Service Pack 4.
		
		`lib/solaris/mod_wl_ssl_raven.so`
		For Apache+FastStart installations the `mod_wl_ssl_raven.so` is used.
		The `mod_wl_ssl_raven.so` file was required because FastStart applies frontpage patches which makes it incompatible with `mod_wl_ssl.so`. It is available beginning with WLS 5.1.0 Service Pack 4)

Solaris Tunable Parameters

Warning: Do not modify these parameters before first consulting your Solaris Administrator.

This section presents suggested values for some Solaris TCP and VM-related tunable parameters.

For more information about Solaris configuration, see the Solaris FAQ: http://www.science.uva.nl/pub/solaris/solaris2/index.html.

Tuning Solaris File Descriptor Limits

On Solaris, each user account has a certain limited number of file descriptors. Use the `ulimit` command to print or set resource limits. A resource limit is a pair of values that specify the current (soft) limit and the maximum (hard) limit. You can modify the hard limit in `/etc/system`. You must reboot your machine anytime you modify `/etc/system`.

Note: Do not change the default soft limit. It has the potential to affect many processes on the server and will not affect WebLogic Server.

You must have adequate permissions to use the `ulimit` command. Any user may lower a hard limit. Only a super-user may raise a hard limit.

Viewing and Adjusting File Descriptor Limits

To view and adjust file descriptor limits:

1. Use the `ulimit` command to print current resource limits.
   ```
   ulimit
   ```
2. Set the hard limit value in /etc/system, according to your needs. For example:
   ```
   set rlim_fd_max=4096 /* hard limit */
   ```
3. Restart WebLogic Server. A message similar to the following appears in the startup log:
   ```
   <Jul 30, 2001 9:16:11 AM EDT> <Info> <Posix Performance Pack> <System
   has file descriptor limits of - soft: '256', hard: '4096'>
   <Jul 30, 2001 9:16:11 AM EDT> <Info> <Posix Performance Pack> <Using
   effective file descriptor limit of: '4096' open sockets/files.>
   ```

Setting TCP Parameters with the ndd Command

Set the following TCP tuning parameters with the `ndd` command. Use the following syntax:

```
ndd -set driver parameter
```

For example:

```
ndd -set /dev/ip ip_ignore_redirect 1
```

Table 15-1 Suggested TCP Parameter Values

Parameter	Suggested Value
/dev/ip ip_ignore_redirect	1
/dev/tcp tcp_conn_grace_period	500
/dev/tcp tcp_conn_req_max_q	8096 (*)
/dev/tcp tcp_conn_req_max_q0	8096 (*)
/dev/tcp tcp_conn_req_min	1
/dev/tcp tcp_cwnd_max	65534
/dev/tcp tcp_fin_wait_2_flush_interval	16000
/dev/tcp tcp_ip_abort_cinterval	60000
/dev/tcp tcp_ip_abort_interval	60000
/dev/tcp tcp_keepalive_interval	90000
/dev/tcp tcp_recv_hiwat	32768
/dev/tcp tcp_rexmit_interval_initial	3000
/dev/tcp tcp_rexmit_interval_max	10000
/dev/tcp tcp_rexmit_interval_min	3000
/dev/tcp tcp_slow_start_initial	2
/dev/tcp tcp_time_wait_interval	60000 (**)
/dev/tcp tcp_xmit_hiwat	32768

Set parameters followed with an asterisk based on the number of concurrent connections against the web server.

The `tcp_close_wait_interval` parameter determines the time interval that a TCP socket is kept alive after issuing a close call. The default value of this parameter on Solaris is four minutes.

When a large number of clients connect for a short amount of time, holding these socket resources can have a significant negative impact on performance. Setting this parameter to a value of 60000 (60 seconds) has shown a significant throughput enhancement when running benchmark JSP tests on Solaris.

You might want to reduce this setting further if the server gets backed up with a queue of half-opened connections.

Viewing TCP Parameters with the netstat -sP tcp Command

Use `netstat -sP tcp` to view the following parameters:

Table 15-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpActiveOpens	413995
tcpAttemptFails	15937
tcpCurrEstab	13
tcpEstabResets	621691
tcpHalfOpenDrop	0
tcpInAckBytes	4290207863
tcpInAckSegs	43347796
tcpInAckUnsent	0
tcpInClosed	1700
tcpInDupAck	1230270
tcpInDupBytes	12551831
tcpInDupSegs	7023
tcpInInorderBytes	3228433225
tcpInInorderSegs	42175934
tcpInPartDupBytes	4690354

Table 15-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpInPartDupSegs	7023
tcpInPastWinBytes	486243985
tcpInPastWinSegs	178
tcpInSegs	64462930
tcpInUnorderBytes	258926562
tcpInUnorderSegs	206376
tcpInWinProbe	7584
tcpInWinUpdate	2754
tcpListenDrop	123865 At run time, if `tcpListenDrop` is non-zero and increasing, use the `ndd` command to increase `tcp_conn_req_max_q`.
tcpListenDropQ0	0 If `tcpListenDropQ0` is non-zero and increasing, use the `ndd` command to increase `tcp_conn_req_max_q0`.
tcpMaxConn	-1
tcpOutAck	12416786
tcpOutAckDelayed	3223013
tcpOutControl	4296229
tcpOutDataBytes	4290849539
tcpOutDataSegs	55577630
tcpOutFastRetrans	42726
tcpOutRsts	2735468

Table 15-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpOutSackRetrans	223
tcpOutSegs	68076290
tcpOutUrg	221
tcpOutWinProbe	4193
tcpOutWinUpdate	13576
tcpPassiveOpens	655052
tcpRetransBytes	148287157
tcpRetransSegs	242760
tcpRtoAlgorithm	4
tcpRtoMax	60000
tcpRtoMin	200
tcpRttNoUpdate	95781
tcpRttUpdate	42878526
tcpTimKeepalive	12798
tcpTimKeepaliveDrop	13
tcpTimKeepaliveProbe	2408
tcpTimRetrans	150215
tcpTimRetransDrop	526

Setting tcp_conn_hash_size Parameter in /etc/system

You must reboot your machine anytime you modify `/etc/system` parameters. Set the value of the `tcp_conn_hash_size` parameter in the `/etc/system` file:

```
set tcp:tcp_conn_hash_size=8192
```

VM Parameters

On JDK 1.2.2_05a, set the following parameter in the java command line:

```
java <other command line options>
-Xgenconfig:64m,64m,semispaces:64m,512m,markcompact weblogic.Server
```

See also “Tuning Java Virtual Machines” in the BEA WebLogic Server Performance and Tuning guide:

For a detailed explanation of the JDK 1.3 VM, see:

http://java.sun.com/j2se/docs/PerformanceFAQ.html

http://java.sun.com/j2se/docs/VMOptions.html

CHAPTER 16

Sun Microsystems/Fujitsu SPARC with Solaris 8

General Information

- To download supported releases for this platform, see Download WebLogic Server for Your Platform.
- WebLogic supports Sun Solaris on Sun Microsystems/Fujitsu SPARC computers. For information on Solaris, see Sun's Solaris web page.
- For Sun's support services info site, see http://www.sun.com/service/support. If you have a support contract for the use of Solaris platforms, you may be able to obtain support for general Java issues (for example, JVM problems).
- You can run WebLogic software on Solaris using the SDK available from JavaSoft. BEA recommends using the latest BEA supported version of the JVM's available, except where noted on our Supported Configurations document.
- The JIT (Just-In-Time) Compiler is only supported for 1.2.2_05a while using the `export _JVM_ARGS="inline_instrs_jit=0"` directive.This is due to Sun's bugs: #4333396 (JVM crash during JIT) and #4333696. The error message reads: `SIGABRT and JIT coredump: panic: Unexpected operand kind(10) in j86OpndImage`. These also affect 1.2.1_04, so that you must Disable aggressive inlining by the jit with: `export _JIT_ARGS=novinline.`
- 7.3.4 jDriver for Oracle is not supported with JDK 1.3.1.
- We have encountered a problem where some Solaris JVMs report an incorrect IP number. The reported IP number for the machine is the first number in the `/etc./hosts` file, which, unfortunately, is frequently `127.0.0.1`, the standard loopback network address for

the machine. This problem affects licenses, where a license is granted for a specific IP number. We have also observed this problem in a cluster where an HTTP request is passed by a web server or another WebLogic Server via the HttpClusterServlet, to a WebLogic Server running on a Solaris machine with this configuration problem. The `127.0.0.1` IP number is carried with the server's response. Subsequent calls to that machine time out because of the incorrect IP number. You can run the command `java utils.netAddresses` to see if you have this problem on your Solaris computer. If you see `127.0.0.1` in the output, ask the system administrator to move the computer's network IP address to the beginning of the `/etc/hosts` file.

- For information on suggested values for some Solaris TCP and VM-related tunable parameters, see Solaris Tunable Parameters

Supported Releases

- WebLogic 5.1

Known JDK Issues

Problems with JDK 1.3 crashing

If you have problems with OutOfMemory errors and the JVM crashing with JDK 1.3, try setting: -XX:MaxPermSize=128m. There is currently an open bug on Sun's bug parade that describes this problem. See http://developer.java.sun.com/developer/bugParade/bugs/4390238.html.

JDK 1.2.2_06

BEA recommends using the 1.2.2_05a version with the directive `export _JVM_ARGS="inline_instrs_jit=0".` This will disable aggressive inlining by the JIT while allowing it to perform other performance enhancing tasks. The 1.2.2_05a version has been supported by BEA. There is a known garbage collection related deadlock condition, bug #4351991, with JDK 1.2.2_06. If you wish to use the 1.2.2_06 version, emergency relief that fixes bug #4351991 will be available through Sun, up until the release of 1.2.2_07 or the full BEA qualification of a later Sun JDK. Please contact your Sun support representative with any questions regarding this relief.

JDK 1.2.2_05a

BEA recommends using the 1.2.2_05a version with the directive `export _JVM_ARGS="inline_instrs_jit=0"`. This disables aggressive inlining by the JIT while allowing it to perform other performance enhancing tasks.

JDK 1.2.2

The wrapper script for the Solaris JDK 1.2.2 ignores the value of `$THREADS_FLAG` and defaults to using green threads unless you supply the `-native` option. WebLogic Server is not supported with green threads. Please specify the `-native` option on all of your `java` commands.

JDK 1.2.1_04

You must disable aggressive inlining by the JIT with: `export _JIT_ARGS=novinline`

WebLogic 5.1

Operating System	Solaris 8
Required OS Patches	None
CPU	UltraSPARC 168MHz or better
RAM	64 MB minimum
Disk Space	256 MB

Supported Client JVM	• SunSoft SDK 1.3.1_06 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_06-b01) Java HotSpot™ Client VM (build 1.3.1_06-b01, mixed mode) • SunSoft SDK1.3.1_04 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_04-b02) Java HotSpot™ Client VM (build 1.3.1_04-b02, mixed mode) • SunSoft SDK 1.3.1_02 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_02-b02) with Java HotSpot™ Client VM (build 1.3.1_02-b02, mixed mode) • SunSoft SDK 1.3.1_01 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_01), Java HotSpot™ Client VM (build 1.3.1_01, mixed mode) • SunSoft SDK 1.3.1 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1-b24) with Java HotSpot™ Client VM (build 1.3.1-b24, mixed mode)	WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1.

Supported Client JVM, continued	• SunSoft SDK 1.3.0 Java™ 2 SDK Runtime Environment, Standard Edition (build 1.3.0), with Java HotSpot™ Client VM (build 1.3.0, mixed mode) • JDK 1.2.2_09 (build Solaris_JDK_1.2.2_09, native threads, sunwjit) • JDK 1.2.2_07 (build Solaris_JDK_1.2.2_07, native threads, sunwjit) • JDK 1.2.2_06 See BEA recommendations at JDK 1.2.2_06. • JDK 1.2.2_05a See BEA recommendations at JDK 1.2.2_05a. • SunSoft JDK 1.2.1_04 See BEA recommendations at JDK 1.2.1_04.

Supported Server JVM	• SunSoft SDK 1.3.1_06 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_06-b01) Java HotSpot™ Client VM (build 1.3.1_06-b01, mixed mode) • SunSoft SDK1.3.1_04 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_04-b02) Java HotSpot™ Client VM (build 1.3.1_04-b02, mixed mode) • SunSoft SDK 1.3.1_02 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_02-b02) with Java HotSpot™ Client VM (build 1.3.1_02-b02, mixed mode) • SunSoft SDK 1.3.1_01 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1_01), Java HotSpot™ Client VM (build 1.3.1_01, mixed mode)	WebLogic 5.1 was certified with SunSoft SDK 1.3.1 and will support any service pack of SDK 1.3.1.

Supported Server JVM, continued	• SunSoft SDK 1.3.1 Java™ 2 Runtime Environment, Standard Edition (build 1.3.1-b24) with Java HotSpot™ Client VM (build 1.3.1-b24, mixed mode) • SunSoft SDK 1.3.0 Java™ 2 SDK Runtime Environment, Standard Edition (build 1.3.0), with Java HotSpot™ Client VM (build 1.3.0, mixed mode) • JDK 1.2.2_09 (build Solaris_JDK_1.2.2_09, native threads, sunwjit) • JDK 1.2.2_07 (build Solaris_JDK_1.2.2_07, native threads, sunwjit) • JDK 1.2.2_06 See BEA recommendations at JDK 1.2.2_06. • JDK 1.2.2_05a See BEA recommendations at JDK 1.2.2_05a. • SunSoft JDK 1.2.1_04 See BEA recommendations at JDK 1.2.1_04.	
Performance Pack	Included.	
Node Manager	None.	
Oracle jDriver	**For this Oracle Server version:** Oracle 8.1.7 Requires Service Pack 10 or higher	**This Oracle jDriver shared library is supported:** `lib/solaris/oci817_8/libweblogicoci37.so`

NSAPI Plug-ins	**For this plugin:**	**This shared library is supported:**
	iPlanet 3.6	`lib/solaris/libproxy.so`
	iPlanet 4.0	`lib/solaris/libproxy.so`
	iPlanet 4.1.x	`lib/solaris/libproxy.so`
	Sun One 6.0 WebLogic 5.1 Service Pack 11 and later	`lib/solaris/libproxy.so`
Apache Plug-ins	**For this Apache version:**	**These shared libraries are supported:**

	Apache 1.3.x	`lib/solaris/mod_wl.so`
		The `mod_wl.so` is a shared object for standard Apache (non EAPI).
		
		`lib/solaris/mod_wl_ssl.so`
		The file `mod_wl_ssl.so` is used for Apache + SSL/EAPI installations (Stronghold, modssl etc). It is available beginning with WLS 5.1.0 Service Pack 4.
		
		`lib/solaris/mod_wl_ssl_raven.so`
		For Apache+FastStart installations the `mod_wl_ssl_raven.so` is used.
		The `mod_wl_ssl_raven.so` file was required because FastStart applies frontpage patches which makes it incompatible with `mod_wl_ssl.so`. It is available beginning with WLS 5.1.0 Service Pack 4)

Solaris Tunable Parameters

Warning: Do not modify these parameters before first consulting your Solaris Administrator.

This section presents suggested values for some Solaris TCP and VM-related tunable parameters.

For more information about Solaris configuration, see the Solaris FAQ: http://www.science.uva.nl/pub/solaris/solaris2/index.html.

Tuning Solaris File Descriptor Limits

On Solaris, each user account has a certain limited number of file descriptors. Use the `ulimit` command to print or set resource limits. A resource limit is a pair of values that specify the current (soft) limit and the maximum (hard) limit. You can modify the hard limit in `/etc/system`. You must reboot your machine anytime you modify `/etc/system`.

Note: Do not change the default soft limit. It has the potential to affect many processes on the server and will not affect WebLogic Server.

You must have adequate permissions to use the `ulimit` command. Any user may lower a hard limit. Only a super-user may raise a hard limit.

Viewing and Adjusting File Descriptor Limits

To view and adjust file descriptor limits:

1. Use the `ulimit` command to print current resource limits.
   ```
   ulimit
   ```
2. Set the hard limit value in /etc/system, according to your needs. For example:
   ```
   set rlim_fd_max=4096 /* hard limit */
   ```
3. Restart WebLogic Server. A message similar to the following appears in the startup log:
   ```
   <Jul 30, 2001 9:16:11 AM EDT> <Info> <Posix Performance Pack> <System
   has file descriptor limits of - soft: '256', hard: '4096'>
   <Jul 30, 2001 9:16:11 AM EDT> <Info> <Posix Performance Pack> <Using
   effective file descriptor limit of: '4096' open sockets/files.>
   ```

Setting TCP Parameters with the ndd Command

Set the following TCP tuning parameters with the `ndd` command. Use the following syntax:

```
ndd -set driver parameter
```

For example:

```
ndd -set /dev/ip ip_ignore_redirect 1
```

Table 16-1 Suggested TCP Parameter Values

Parameter	Suggested Value
/dev/ip ip_ignore_redirect	1
/dev/tcp tcp_conn_grace_period	500
/dev/tcp tcp_conn_req_max_q	8096 (*)
/dev/tcp tcp_conn_req_max_q0	8096 (*)
/dev/tcp tcp_conn_req_min	1
/dev/tcp tcp_cwnd_max	65534
/dev/tcp tcp_fin_wait_2_flush_interval	16000
/dev/tcp tcp_ip_abort_cinterval	60000
/dev/tcp tcp_ip_abort_interval	60000
/dev/tcp tcp_keepalive_interval	90000
/dev/tcp tcp_recv_hiwat	32768
/dev/tcp tcp_rexmit_interval_initial	3000
/dev/tcp tcp_rexmit_interval_max	10000
/dev/tcp tcp_rexmit_interval_min	3000
/dev/tcp tcp_slow_start_initial	2
/dev/tcp tcp_time_wait_interval	60000 (**)
/dev/tcp tcp_xmit_hiwat	32768

Set parameters followed with an asterisk based on the number of concurrent connections against the web server.

For Solaris 2.5.1 and 2.6, `tcp_time_wait_interval` has changed to:

```
ndd -set /dev/tcp tcp_close_wait_interval 60000
```

The `tcp_close_wait_interval` parameter determines the time interval that a TCP socket is kept alive after issuing a close call. The default value of this parameter on Solaris is four minutes. When a large number of clients connect for a short amount of time, holding these socket resources can have a significant negative impact on performance. Setting this parameter to a value of 60000 (60 seconds) has shown a significant throughput enhancement when running benchmark JSP tests on Solaris.

You might want to reduce this setting further if the server gets backed up with a queue of half-opened connections.

Viewing TCP Parameters with the netstat -sP tcp Command

Use `netstat -sP tcp` to view the following parameters:

Table 16-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpActiveOpens	413995
tcpAttemptFails	15937
tcpCurrEstab	13
tcpEstabResets	621691
tcpHalfOpenDrop	0
tcpInAckBytes	4290207863
tcpInAckSegs	43347796
tcpInAckUnsent	0
tcpInClosed	1700
tcpInDupAck	1230270
tcpInDupBytes	12551831
tcpInDupSegs	7023
tcpInInorderBytes	3228433225

Table 16-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpInInorderSegs	42175934
tcpInPartDupBytes	4690354
tcpInPartDupSegs	7023
tcpInPastWinBytes	486243985
tcpInPastWinSegs	178
tcpInSegs	64462930
tcpInUnorderBytes	258926562
tcpInUnorderSegs	206376
tcpInWinProbe	7584
tcpInWinUpdate	2754
tcpListenDrop	123865 At run time, if `tcpListenDrop` is non-zero and increasing, use the `ndd` command to increase `tcp_conn_req_max_q`.
tcpListenDropQ0	0 If `tcpListenDropQ0` is non-zero and increasing, use the `ndd` command to increase `tcp_conn_req_max_q0`.
tcpMaxConn	-1
tcpOutAck	12416786
tcpOutAckDelayed	3223013
tcpOutControl	4296229
tcpOutDataBytes	4290849539
tcpOutDataSegs	55577630

Table 16-2 Viewable TCP Parameter Values

Parameter	Suggested Value
tcpOutFastRetrans	42726
tcpOutRsts	2735468
tcpOutSackRetrans	223
tcpOutSegs	68076290
tcpOutUrg	221
tcpOutWinProbe	4193
tcpOutWinUpdate	13576
tcpPassiveOpens	655052
tcpRetransBytes	148287157
tcpRetransSegs	242760
tcpRtoAlgorithm	4
tcpRtoMax	60000
tcpRtoMin	200
tcpRttNoUpdate	95781
tcpRttUpdate	42878526
tcpTimKeepalive	12798
tcpTimKeepaliveDrop	13
tcpTimKeepaliveProbe	2408
tcpTimRetrans	150215
tcpTimRetransDrop	526

Setting tcp_conn_hash_size Parameter in /etc/system

You must reboot your machine anytime you modify `/etc/system` parameters. Set the value of the `tcp_conn_hash_size` parameter in the `/etc/system` file:

```
set tcp:tcp_conn_hash_size=8192
```

VM Parameters

On JDK 1.2.2_05a, set the following parameter in the java command line:

```
java <other command line options>
-Xgenconfig:64m,64m,semispaces:64m,512m,markcompact weblogic.Server
```

See also "Tuning Java Virtual Machines" in the BEA WebLogic Server Performance and Tuning guide:

For a detailed explanation of the JDK 1.3 VM, see:

- http://java.sun.com/j2se/docs/PerformanceFAQ.html
- http://java.sun.com/j2se/docs/VMOptions.html

CHAPTER 17

Installation Help for Your Platform

This document contains installation instructions for the following platforms:

- Compaq Open VMS
- IBM OS/400 V5R1
- IBM OS/390 Platform Support

Compaq Open VMS

For general information about Java technology for Compaq Alpha Systems, see the Compaq Java Technology Center website.

- Information on Compaq VMS with WebLogic Server 5.1

Information on Compaq VMS with WebLogic Server 5.1

OpenVMS v7.2 has been tested on WebLogic Server 5.1.with jdbcKona/MSSQLServer4. Note the following, however:

WebLogic Server must be started statically for the OpenVMS platform.

Make sure `compileCommand=javac` is valued under your `weblogic.httpd.register.*.jsp` entry in the weblogic properties file. This will enable compilation of JSP's on the OpenVMS platform.

Known Issues with OpenVMS v7.2 on WebLogic 5.1

`ActivatableRuntimeDescriptor.getStub()` is not supported on this platform.

IBM OS/400 V5R1

Known Issues

There are a number of known issues associated with this release on this platform:

- Currently, neither the AS/400 native driver nor the toolbox JDBC driver support `TMJOIN` and `TMRESUME` in `javax.transaction.xa.XAResource`. The AS/400 XA JDBC driver doesn't work with WebLogic Server's JTA transaction manager.
- WebLogic's NodeManager function is not supported in the AS/400 release.
- The WebLogic installer for AS/400 doesn't contain a WebLogic evaluation license, please contact BEA Sales to get an evaluation license.
- The initial system password of WebLogic Servers is `abcd1234`. You should change it the first time when server is started.

Installation

Download the WebLogic distribution

For information on how to download this platform, see Download WebLogic Server for Your Platform.

1. Find your WebLogic Server release and Service Pack on the download page.
1. From your client workstation, download WebLogic for AS/400 from the WebLogic web site. (`webl610sp2.zip`)
2. Use a 32-bit program such as WinZip to unpack the zip file.

Transfer WebLogic to the AS/400 and install

1. Log into the AS/400 with an administrative account.
2. Create a save file in the AS/400 library `QGPL`:

```
===> CRTSAVF FILE (QGPL/WEBL610sp2)
```

This command creates a save file on the AS/400 where you will transfer the webl610sp2 file.

3. At a command prompt on the Windows NT client computer, type:

   ```
   $ ftp AS/400-hostname $ bin $ put webl610sp2 QGPL/WEBL610sp2
   ```

4. Log into the AS/400 with an administrative account. Restore the save file:

   ```
   ===> RSTLIB SAVLIB(WEBLOGIC61) DEV(*SAVF)SAVF(QGPL/WEBL610SP2)
   ```

5. Run the WebLogic installation script:

   ```
   ===> WEBLOGIC61/INSWEBL
   ```

   The `INSWEBL` script does the following:

   a. Creates `WEBLOGIC` and `WLADMIN` user profiles.

   b. Gives `WEBLOGIC` and `WLADMIN` user profiles all authority for all commands in the `WEBLOGIC61` library.

   c. Installs WebLogic and the Java programs associated with its classes, .jar files, and .zip files. During this process, you will see the message:

      ```
      Restoring WebLogic Server files. Please wait.
      ```

      If the restore succeeds, you will see the message:

      ```
      WebLogic Server product successfully installed.
      ```

      If the restore does not succeed, you will see the message

      ```
      WebLogic Server product did not install successfully.
      ```

6. If you want, you can delete the save file with the command:

   ```
   ===> DLTF FILE(QGPL/WEBL610sp2)
   ```

   On your Windows NT workstation, you can also delete the .zip file you downloaded and the `webl610sp2` file.

7. The `WEBLOGIC61/INSWEBL` programs `(*PGM)` and command `(*CMD)` are shipped with public authority set to `*USE`. Once the install has been run successfully, the administrator may set the authority on these objects to `PUBLIC *EXCLUDE`.

   Your AS/400 now has a WebLogic `IFS` directory containing the WebLogic Java classes as well as configuration files and example programs.

   The Java classes are transformed with the AS/400 Java transformer with level 40 optimization, so AS/400 direct executables for WebLogic are used.

8. The weblogic server is installed at /weblogic/610sp2. The BEA home directory is /weblogic/61sp2. The WebLogic product installation directory is /weblogic/61sp2/wlserver6.1.

   For more information about the BEA home directory, please refer to the discussion here:

   http://e-docs.bea.com/wls/docs61/install/instpre.html#1066128

Add WebLogic administrative users

WebLogic administrative users have *ALL authority for the commands in the WEBLOGIC61 library and *RWX authority on certain files in the /weblogic directory. To allow a user to start and stop the server or to modify any of these files, you can add GRPPRF(WLADMIN) to their profiles.

To create a new user with authority to administer WebLogic, use this command:

```
===> CRTUSRPRF USRPRF(USERNAME) GRPPRF(WLADMIN)
```

To add authority to administer WebLogic to an existing user, use this command:

```
===> CHGUSRPRF USRPRF(USERNAME) GRPPRF(WLADMIN)
```

About the WEBLOGIC user

The WEBLOGIC61/INSWEBL command creates the WEBLOGIC user profile with authorizations of *NONE. You cannot log in as WEBLOGIC. However, you can add users to the WLADMIN group, allowing them to start WebLogic Server.

About the evaluation license

This installer can not generate evaluation license dynamically. Please contact BEA customer support to download an evaluation license at the BEA product download site.

Setting the CLASSPATH and other environment variables

Before you start WebLogic Server, you must add the WebLogic classes to the CLASSPATH environment variable.

The CLASSPATH environment variable for the session in which you run WebLogic Server must contain these classes:

```
/weblogic/610sp2/wlserver6.1/lib/os400/wlos400.jar:/weblogic/610sp2/wls
erver6.1/lib/weblogic.jar
```

You must also set the QIBM_MULTI_THREADED environment variable to Y before you run STRWEBL, JAVA, or RUNJAVA commands or the JAVA command in an interactive QSHELL shell.

You can enter the CLASSPATH as a parameter when you run the STRWEBL CL command. Or you can use ADDENVVAR to add variables to your environment, CHGENVVAR to change an environment variable, or WRKENVVAR to add or modify variables in the environment.

The next sections describe how to set the CLASSPATH and other variables on the AS/400 using ADDENVVAR or QSHELL and on a Windows NT workstation. (The commands are displayed on several lines here for clarity, but they should be typed without line breaks when you enter them.)

Setting environment variables with the CL ADDENVVAR command

The following commands set the CLASSPATH and QIBM_MULTI_THREADED environment variables in an AS/400 session:

```
===> ADDENVVAR ENVVAR(CLASSPATH) VALUE(
'/weblogic/610sp2/wlserver6.1/lib/os400/wlos400.jar:/weblogic/610sp2/wl
server6.1/lib/weblogic.jar')

===> ADDENVVAR ENVVAR(QIBM_MULTI_THREADED) VALUE('Y')
```

Setting environment variables in QSHELL

If you set environment variables in your session before you start QSHELL, the variables remain set in QSHELL, unless a ".profile" file in your home directory or the /etc/profile file resets environment variables.

You can set environment variables for every interactive QSHELL shell you start with STRQSH or QSH by setting the CLASSPATH in the .profile file in your home directory. An administrator can set environment variables globally by adding them to the /etc/profile file, which is executed whenever any user starts a QSHELL shell with the STRQSH or QSH commands.

To set environment variables in QSHELL, use these commands at the prompt, or add them to one of the profile files:

```
$ export -s \
CLASSPATH='/weblogic/610sp2/wlserver6.1/lib/os400/wlos400.jar:/weblogic
/610sp2/wlserver6.1/lib/weblogic.jar')

$ export -s QIBM_MULTI_THREADED=Y
```

Setting CLASSPATH on a workstation with Client Access

You must set the CLASSPATH environment variable on a workstation running WebLogic classes from the AS/400. The QIBM_MULTI_THREADED environment variable is needed only when you are running Java classes in the AS/400 VM.

In this example, AS400SRVR is the name of the AS/400 server where WebLogic Server is installed.

```
C:\>SET \
CLASSPATH==%CLASSPATH%://AS400SRVR/weblogic/610sp2/wlserver6.1/lib/os40
0/wlos400.jar://AS400SRVR/weblogic/610sp2/wlserver6.1/lib/weblogic.jar
```

Starting WebLogic Server

The WEBLOGIC61/STRWEBL command starts WebLogic Server. This command changes to the WEBLOGIC account before it starts the server. This is important because objects the server creates belong to the WEBLOGIC account. If you start the server using a method other than the WEBLOGIC/STRWEBL command, you may experience problems with permissions. With QSHELL you can start WebLogic sever using the shell script shipped with WebLogic server, but WebLogic Server will be using your user profile. This could be a security problem in a production environment.

To start WebLogic Server with the WEBLOGIC61/STRWEBL command:

1. Log into the AS/400 with an account that is a member of the WLADMIN group.
2. Change to the WebLogic root directory. For example, there is a default root directory shipped with WebLogic server: "/weblogic/610sp2/wlserver6.1".

   ```
   ===> CD '/weblogic/610sp2/wlserver6.1'
   ```
3. Make sure that the CLASSPATH variable is set as described in the previous section.
4. Make sure there is no active QSHELL session within the current job. You should press F3 to quit a QSHELL session, otherwise WebLogic Server won't start.
5. Start WebLogic Server with the following CL command:

   ```
   ===> WEBLOGIC61/STRWEBL
   ```
6. Enter domain name in the WebLogic Domain Name field. For example, enter mydomain (the default shipped with Weblogic)
7. Enter server name in the WebLogic Server Name field. For example, enter myserver (the default shipped with WebLogic).
8. In the bea.home Directory field, enter the bea home directory, or use *ROOT_INSTALL_DIR as default, which will point to /weblogic/610sp2

9. In the “WebLogic root directory” field, enter the default value `*CURDIR` (which will point to `/weblogic/610sp2/ wlserver6.1`).
   Optionally, enter the directory that contains the security resources for your domains and your configuration files.
10. Enter `CLASSPATH` in the Classpath field or use `*ENVVAR` as default.
11. Enter JVAV and WebLogic properties and values in the "Property name" and "Property value" list.

    For more information about WebLogic Server startup properties, see Starting and Stopping WebLogic Servers.
12. If you want to run WebLogic Server in the background, change the Submit field to `*YES`.
13. Modify any of the other parameters as needed.

    For parameters such as `CLASSPATH`, there may not initially be enough room to enter the entire value. You can expand a line by entering an ampersand (&) before you enter the value. Keep entering ampersands until there is enough room to enter your value. Press `Enter` to execute the command.

    For example, the following command starts the default server shipped with WebLogic Server:

    ```
    ===> WEBLOGIC61/STRWEBL DOMAINNAME(mydomain) SERVERNAME(myserver)
    PROPERTIES((java.security.policy
    '=/weblogic/610sp2/wlserver6.1/lib/weblogic.policy'))
    ```

    If you start the WebLogic Server interactively (Submit set to `*NO`), do not press F3 or F12 to exit the Java Command Entry Screen until you see a message similar to this:

    ```
    Java program completed
    ```

    See the WebLogic Server Administration Guide for detailed instructions on administering WebLogic, including setting up WebLogic Server as an HTTP server.

    WebLogic Server has extensive properties to give you a fine degree of control its operation.

    Also read about the property-related changes in the release notes for WebLogic Server.

Stopping WebLogic Server

You can shut down WebLogic Server through the Administration Console, or enter the `ENDWEBL` command:

```
===> WEBLOGIC61/ENDWEBL
```

Enter the WebLogic administrative user name (usually `system`) and password, and then press `Enter` to continue. Wait until you see a message similar to the following:

```
Shutdown initiated
The shutdown sequence has been initiated.
Java program completed
```

After this message appears, you can press F3 to exit the Java Command Entry Screen and return to the AS/400 Command Entry screen.

First time Set-up for the default servers shipped with WebLogic.

BEA ships three default servers with WebLogic product. The configuration repository can be found at `/weblogic/610sp2/wlserver6.1/config`.

Server `myserver` in Domain `mydomain` is the default product server.

The initial system password of WebLogic Servers is `abcd1234`. You should change it the first time when server is started.

To change the password:

1. Start your default WebLogic Server. When the system asks for your password, type:

   ```
   abcd1234
   ```

2. After the server started, open an Administration Console from a web browser using the URL:

   `http://AS400NAME:PORT/console`

   Enter `system` as your administrative user name and `abcd1234` as your password.

3. In the left panel of the console, click on the Users icon under the Security category.

4. Change the system password on the right panel.

Additional demo domains

Two additional demo domains are also shipped: PetStore and Examples.

These servers were supposed to use the evaluation copy of CloudScape database shipped with WebLogic. We found Cloudscape doesn't work with JDK1.3.1 on AS/400, so a DDL for demo database is shipped with WebLogic.

To set up the demo database, execute the following `CL` command:

```
RUNSQLSTM SRCFILE(WEBLOGIC61/SQLSRC) SRCMBR(WLSDEMO) COMMIT(*NONE)
```

A demo database named WLSDEMO will be created on your machine. If you want to use a different name for the database, please edit member WLSDEMO in the file WEBLOGIC61/SQLSRC, replacing the string wlsdemo with the name you chose.

To set up proper JDBC connection pool properties like user name and password, you also need to modify file config.xml in these directories:

- /weblogic/610sp2/wlserver6.1/config/petstore
- /weblogic/610sp2/wlserver6.1/config/examples

For more information about setting up JDBC connections please refer to the WebLogic for IBM AS/400e Technical FAQ.

Starting the PetStore server

1. Modify /weblogic/610sp2/wlserver6.1/config/petstore/config.xml, seting user id and password for the JDBC connection pool to demo database WLSDEMO.
2. Execute:

   ```
   ===> CD '/weblogic/610sp2/wlserver6.1'
   ```

3. Set CLASSPATH to:

   ```
   /weblogic/610sp2/wlserver6.1/lib/os400/wlos400.jar:/weblogic/610sp2/wls
   erver6.1/lib/weblogic.jar:/weblogic/610sp2/wlserver6.1/config//petstore
   /serverclasses
   ```

4. The initial password for this server is abcd1234
5. Execute:

   ```
   ===> WEBLOGIC61/STRWEBL DOMAINNAME(petstore) SERVERNAME(petstoreServer)
   PROPERTIES(
   (java.security.policy'=/weblogic/610sp2/wlserver6.1/lib/weblogic.policy
   '))
   ```

Starting the Examples Server

1. Modify /weblogic/610sp2/wlserver6.1/config/examples/config.xml, set up user name and password for the JDBC connection pool to demo database WLSDEMO.
2. Execute:

   ```
   ===> CD '/weblogic/610sp2/wlserver6.1'
   ```

3. set CLASSPATH to

```
/weblogic/610sp2/wlserver6.1/lib/os400/wlos400.jar:/weblogic/610sp2/wls
erver6.1/lib/weblogic.jar
```

4. The initial password for this server is `abcd1234`
5. Execute:

```
===> WEBLOGIC61/STRWEBL DOMAINNAME(examples) SERVERNAME(examplesServer)
PROPERTIES((java.security.policy
'=/weblogic/610sp2/wlserver6.1/lib/weblogic.policy'))
```

WebLogic Developer Center for IBM AS/400e

The AS/400 Developer Center is your starting point for using WebLogic Server on the IBM AS/400. From the Developer Center you can access documentation written specifically about using WebLogic software in an AS/400 environment, as well as documents relevant to all WebLogic users.

- WebLogic Server 5.1 Developer Center for AS/400e:

  http://www.weblogic.com/docs51/platforms/as400/index.html

IBM OS/390 Platform Support

For more supported platform information, see "IBM S/390 with OS/390" on page 8-1.

General Information

For general information about S/390, OS/390, and USS/390, see IBM's S/390 website.

Installation

Install WebLogic Server on OS/390 using the WebLogic .zip archive distribution. Follow the instructions for installing from a .zip archive for your version of WebLogic Server.

Important for WebLogic 5.1 users

You can download the latest JDK that runs on WebLogic Server 5.1 from the S/390 Java page at:

`http://www.s390.ibm.com/java/about118.html`

After you install WebLogic Server on OS/390, you must run the `USS390PostInst` utility, which is in the `util390.jar` file, to convert text files from `ASCII` to `EBCDIC` encoding. It is only necessary to do this one time after you install. Follow these steps:

1. Copy the util390.jar file into your WebLogic home directory.
2. Add `util390.jar` to your CLASSPATH environment variable:

   ```
   $ CLASSPATH=<WebLogic home directory path>/util390.jar:$CLASSPATH;
   export $CLASSPATH
   ```

3. Run the `USS390PostInst` utility:

   ```
   $ java USS390PostInst
   ```

   `USS390PostInst` prints a message as it converts each text file in the WebLogic distribution.

Notes on using WebLogic Server with OS/390

- No native libraries are provided for OS/390.
- Graphical utilities, such as the WebLogic Console, must be executed on a client computer, since USS390 does not have a GUI environment.

  These and other DB2 problems will be addressed in the upcoming Version 8 release of DB2.

Errata

This section records information about important changes to the Weblogic Server Supported Configurations document. These changes may occur for many reasons, such as correction of errors, changes in vendor support, or changes in BEA support. If you have any questions regarding the following information, please contact your BEA Customer Support Representative.

Date	Description
8/15/2003	BEA has learned that a vendor OS patch is required for HP JDK 1.3.1.10. The HP 1.3.1 JVM policy for WebLogic Server 5.1 releases certified with a service pack earlier than 1.3.1.10 was updated to limit support though JDK 1.3.1.09.
04/03/2003	Removed support table for WebLogic 5.1 for AIX 5.1 with IBM JDK 1.3.1. This configuration is not supported by BEA.