



# BEA WebLogic Server® and WebLogic Express®

## Release Notes

Version 8.1  
Revised: April 25, 2006

# Copyright

Copyright © 2005 BEA Systems, Inc. All Rights Reserved.

## Restricted Rights Legend

This software and documentation is subject to and made available only pursuant to the terms of the BEA Systems License Agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the agreement. This document may not, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from BEA Systems, Inc.

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the BEA Systems License Agreement and in subparagraph (c)(1) of the Commercial Computer Software-Restricted Rights Clause at FAR 52.227-19; subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, subparagraph (d) of the Commercial Computer Software--Licensing clause at NASA FAR supplement 16-52.227-86; or their equivalent.

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA Systems DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

## Trademarks or Service Marks

BEA, BEA JRockit, BEA Liquid Data for WebLogic, BEA WebLogic Server, Built on BEA, Jolt, JoltBeans, SteelThread, Top End, Tuxedo, and WebLogic are registered trademarks of BEA Systems, Inc. BEA AquaLogic, BEA AquaLogic Data Services Platform, BEA AquaLogic Enterprise Security, BEA AquaLogic Service Bus, BEA AquaLogic Service Registry, BEA Builder, BEA Campaign Manager for WebLogic, BEA eLink, BEA Manager, BEA MessageQ, BEA WebLogic Commerce Server, BEA WebLogic Enterprise, BEA WebLogic Enterprise Platform, BEA WebLogic Enterprise Security, BEA WebLogic Express, BEA WebLogic Integration, BEA WebLogic Java Adapter for Mainframe, BEA WebLogic JDriver, BEA WebLogic JRockit, BEA WebLogic Log Central, BEA WebLogic Personalization Server, BEA WebLogic Platform, BEA WebLogic Portal, BEA WebLogic Server Process Edition, BEA WebLogic WorkGroup Edition, BEA WebLogic Workshop, and Liquid Computing are trademarks of BEA Systems, Inc. BEA Mission Critical Support is a service mark of BEA Systems, Inc. All other company and product names may be the subject of intellectual property rights reserved by third parties.

All other trademarks are the property of their respective companies.

# Contents

## 1. WebLogic Server 8.1 Features and Changes

What's New in WebLogic Server 8.1 SP5. . . . .	1-2
Enhanced Support for Oracle RAC . . . . .	1-3
XA Call Retry for Oracle RAC . . . . .	1-3
Using MultiPools with Global Transactions . . . . .	1-3
Administration Console Changes . . . . .	1-3
DISCOVERMANAGEDSERVER . . . . .	1-3
SocketBufferSizeAsChunkSize . . . . .	1-4
Optimistic Concurrency for EJBs . . . . .	1-4
HTTP GET Request Limit. . . . .	1-4
Bypassing Compilation Errors in JSP . . . . .	1-4
J-Integra Version. . . . .	1-4
RSA Libraries . . . . .	1-4
Form-based Authentication over HTTPS . . . . .	1-5
Limiting Number of Concurrent Requests for a Session. . . . .	1-5
Option for Warning Message BEA-012034. . . . .	1-5
Accessing the Java <code>weblogic.marathon.ddinit.EarInit</code> Method in a Multi-User Environment 1-5	
Behavior of <code>InitialContext</code> . . . . .	1-5
Custom Classloader Structure with Iterative Development. . . . .	1-6
What's New in WebLogic Server 8.1 SP4. . . . .	1-6
Patch to Provide XA Support for Oracle RAC. . . . .	1-6

Patch Configuration Requirements . . . . .	1-7
JDBC MultiPool Configuration for WebLogic Server with Oracle RAC . . . . .	1-7
Enable XA Call Retry for Oracle RAC . . . . .	1-7
Patch to Support Use with Oracle 9i RAC . . . . .	1-9
Patch Configuration Requirements . . . . .	1-9
Enable XA Call Retry . . . . .	1-9
Target Connection Pools to All Servers . . . . .	1-10
WebLogic Type 4 JDBC Drivers . . . . .	1-11
All Drivers . . . . .	1-11
DB2 Driver . . . . .	1-12
Informix Driver . . . . .	1-12
SQL Server Driver . . . . .	1-12
Oracle Driver . . . . .	1-12
Sybase Driver . . . . .	1-13
EJB Features . . . . .	1-14
Dynamic Finders . . . . .	1-14
PreparedQuery Interface . . . . .	1-14
Security Features . . . . .	1-14
Trust Manager . . . . .	1-14
LDAP . . . . .	1-15
Encryption Utility . . . . .	1-15
Single Pass Identity Assertion . . . . .	1-15
Servlet Features . . . . .	1-15
What's New in WebLogic Server 8.1 SP3 . . . . .	1-15
New SDKs Bundled with WebLogic Platform 8.1 . . . . .	1-16
WebLogic Server Process Edition . . . . .	1-16
Web Services Features . . . . .	1-17
Implementation of the Final 1.0 OASIS Web Services Security Specification . . . . .	1-17

HTTPBindingInfo API For SSL Socket Sharing . . . . .	1-18
New wsdl2Service Attribute to Generate a Skeleton Implementation Class . . . . .	1-18
New servicegen and source2wsdd Attribute to Ignore Authorization Header in SOAP Request . . . . .	1-18
WebLogic XML Digital Signatures API . . . . .	1-18
WebLogic Web Service Security API to Add a Timestamp to a SOAP Message . . . . .	1-18
Change in clientgen Behavior Due to Stricter Compliance With WS-I . . . . .	1-19
JDBC Features . . . . .	1-19
WebLogic Server 8.1SP3 Certified with Oracle 9i RAC on HP-UX . . . . .	1-19
Deprecation of WebLogic jDriver for Oracle . . . . .	1-20
Oracle 10g JDBC Thin Driver . . . . .	1-20
Updated WebLogic Type 4 JDBC Drivers . . . . .	1-20
Multiple JNDI Names for Data Sources . . . . .	1-20
PinnedToThread Connection Pool Property to Increase Performance. . . . .	1-21
JDBC MultiPool Failover Enhancements. . . . .	1-21
JDBC Connection Pool Testing Enhancements. . . . .	1-22
Connection Profiling for Troubleshooting Transaction Remnants on a JDBC Connection 1-22	
Support for XAResource Transaction Timeout . . . . .	1-23
SQL Statement Timeout Enhancements for Pooled JDBC Connections . . . . .	1-24
WebDAV Support . . . . .	1-24
Security Features . . . . .	1-25
WAPEnabled Attribute for HTTP Sessions . . . . .	1-26
WebLogic Tuxedo Connector (WTC) Features. . . . .	1-26
What's New in WebLogic Server 8.1 SP2. . . . .	1-27
Patch Available for Sun Java 2 1.4.2 SDK, Oracle 10g Driver, and SQL Server, Sybase or DB2 Database Users . . . . .	1-27
Configurable Directory for Custom Security Provider MBean Types . . . . .	1-27

EJB Features . . . . .	1-27
EJB 2.0 WebLogic QL Extension for EJB QL Provides upper and lower Functions . . . . .	1-28
EJBGen Distribution and Examples . . . . .	1-28
JDBC Features . . . . .	1-28
BEA WebLogic Type 4 JDBC Drivers for DB2, Informix, Oracle, and Sybase . . . . .	1-28
BEA WebLogic Type 4 JDBC Driver for MS SQL Server Enhancements . . . . .	1-29
Timeouts for JDBC Connection Creation and Statement Execution . . . . .	1-29
Enhancements to the dbping Java Utility . . . . .	1-30
Support for oracleConnection Methods . . . . .	1-30
Autocommit for XA JDBC Drivers . . . . .	1-30
User Configuration and User Key Files for weblogic.Admin and Ant Tasks . . . . .	1-30
viewj and viewj32 Enhancements . . . . .	1-31
WebLogic Web Services Features . . . . .	1-31
source2wsdd Meta-Data Tags . . . . .	1-31
Examples of Using WebLogic Workshop with WebLogic Web Services . . . . .	1-31
What's New in WebLogic Server 8.1 SP1 . . . . .	1-32
Capacity Planning Documentation . . . . .	1-32
JDBC Features . . . . .	1-32
BEA WebLogic Type 4 JDBC Driver for Microsoft SQL Server . . . . .	1-32
JDBC Connection Pool Property to Handle Erroneous Local Transactions . . . . .	1-33
Support for Oracle Virtual Private Databases . . . . .	1-33
New JSP Tag Added to the Console Extension Tag Library . . . . .	1-33
Optimized JMS Topic Subscriber Message Selectors . . . . .	1-34
Specify Non-Anonymous Principals when EJB Container Passivates and Removes Beans1-34	
Web Services Features . . . . .	1-34
Per Operation Message Security . . . . .	1-34
Timestamps In SOAP Messages . . . . .	1-35
Implementing and Assembling a WebLogic Web Service From a Schema File . . . . .	1-35

New Ant Task Features to Support XML Schemas . . . . .	1-35
Mapping Complex Data Types in SOAP Faults to Java Exceptions . . . . .	1-35
Restricting Access to the WebLogic Web Service Home Page . . . . .	1-35
Detailed Overview of Message Security . . . . .	1-36
What's New in WebLogic Server 8.1 . . . . .	1-37
Packaging and Deployment Features . . . . .	1-37
Split Development Directory Structure . . . . .	1-37
Custom Classloading for J2EE Modules . . . . .	1-38
New Application Lifecycle Events . . . . .	1-38
Application-Level Class Library . . . . .	1-38
Dynamic Descriptor Changes . . . . .	1-39
Dynamic Changes to Application-Scoped Pools . . . . .	1-39
enforceClusterConstraints Option Changed . . . . .	1-39
Alternate Deployment Descriptors . . . . .	1-39
No Support for Deploying Unrelated Modules . . . . .	1-39
Improved Deployment Performance and Feedback . . . . .	1-40
Administration Console Deployment Changes . . . . .	1-40
Weblogic.Deployer Changes . . . . .	1-40
Administration Features . . . . .	1-40
JRockit JVM . . . . .	1-41
JRockit Monitoring . . . . .	1-41
Enhanced Configuration Wizard . . . . .	1-41
New Defaults for Development Mode and Production Mode . . . . .	1-41
Message Catalogs . . . . .	1-41
New Logging APIs . . . . .	1-42
Disabling Remote Exceptions . . . . .	1-42
Timer Service . . . . .	1-42
Automatic config.xml Archiving . . . . .	1-43

New Affinity Policies for Java Client Load Balancing . . . . .	1-43
Improved Node Manager. . . . .	1-43
Improved Network Channel Configuration. . . . .	1-43
Changes to Managed Server Independence . . . . .	1-44
Changes to Discovering Managed Servers . . . . .	1-44
JNDI Names in Deployment Descriptors . . . . .	1-45
Administration Console Usability Improvements . . . . .	1-45
weblogic.Admin Features . . . . .	1-46
New Default OOTB Performance Parameter Settings. . . . .	1-47
Changes to Default Execute Queue Names. . . . .	1-47
New Locations of System Administration Documentation . . . . .	1-47
Sample Applications . . . . .	1-49
Security Features. . . . .	1-49
Improved Functionality for Creating Roles and Policies . . . . .	1-49
Improved Support for Keystores and SSL Configuration. . . . .	1-49
Support for the Sun Java Cryptography Extension (JCE) Package . . . . .	1-50
EJB Features and Changes. . . . .	1-50
appc Compiler . . . . .	1-50
Batch Operations. . . . .	1-50
Automatic Table Creation Feature Expanded . . . . .	1-50
Preventing Deadlocks for Container-Managed Persistence Beans . . . . .	1-51
Fine-Grained EJB Redeployment . . . . .	1-51
EJB Query Language (QL) Compiler Enhancements . . . . .	1-51
Performance Improvements. . . . .	1-51
Reloadable EJB Modules . . . . .	1-52
EJB Deployment Assistants . . . . .	1-52
New Default Values for Several EJB-Specific Deployment Descriptor Elements. . . . .	1-52
New dbms-column-type Values. . . . .	1-52

Relationship Caching Functionality Allows Multiple Caching Elements . . . . .	1-53
Disabling Deployment Warning Messages . . . . .	1-53
New Security-related Deployment Descriptor Elements . . . . .	1-53
Constraining Optimistic Row Checking . . . . .	1-54
Storing Table Creation Scripts . . . . .	1-54
Assigning a Message-Driven Bean to a Configured Execute Queue . . . . .	1-54
EJB Features Deprecated in This Release . . . . .	1-55
Administration Console Deployment Descriptor Editor . . . . .	1-55
ejbc Compiler . . . . .	1-55
sql-select-distinct . . . . .	1-55
global-role . . . . .	1-55
run-as-identity-principal . . . . .	1-55
stateless-bean-methods-are-idempotent . . . . .	1-56
J2EE Connector Changes . . . . .	1-56
jCOM Features . . . . .	1-56
JDBC Features and Changes . . . . .	1-57
JDBC Assistants . . . . .	1-57
JDBC Connection Pool Attributes . . . . .	1-57
Enhanced Support for JDBC Extensions . . . . .	1-57
Physical Connection from a Connection Pool . . . . .	1-57
Support for RowSets . . . . .	1-58
Statement Caching . . . . .	1-58
Classes Removed from weblogic.jdbc.pool . . . . .	1-58
JTA Features . . . . .	1-58
Manual Completion of Current Transactions . . . . .	1-58
Non-XA-Compliant Resource Participation in a Global Transaction . . . . .	1-59
JMS Features . . . . .	1-59
JMS Thin Client . . . . .	1-59

Simplified Access to Remote WebLogic Domains or Third-Party JMS Providers . . .	1-59
Easier Access to JMS Via EJBs and Servlets . . . . .	1-59
Better Expired Message Handling . . . . .	1-60
Improved Message Flow Control by Blocking Producers . . . . .	1-60
Ordered Redelivery of Messages . . . . .	1-60
Dynamically Deleting Queue or Topic Destinations . . . . .	1-60
ServerSessionPoolFactory Class Deprecated In weblogic.jms Package . . . . .	1-60
Web Application Features . . . . .	1-61
Performance Improvements . . . . .	1-61
Security Principle for init Method . . . . .	1-61
Allow Access to WEB-INF on forward/include . . . . .	1-61
Integrated jspc Functionality into appc . . . . .	1-61
Class Reloading for Servlet Filters . . . . .	1-61
FileServlet File Sorting Options . . . . .	1-62
Web Services Features . . . . .	1-62
Digital Signatures and Encryption . . . . .	1-62
Reliable SOAP Messaging . . . . .	1-62
SOAP 1.2 . . . . .	1-62
JMS Transport Protocol . . . . .	1-62
Asynchronous Invocation of WebLogic Web Services . . . . .	1-63
Portable Stubs . . . . .	1-63
Implementation of the SOAP With Attachments API For Java (SAAJ) 1.1 . . . . .	1-63
wsdlgen Ant Task . . . . .	1-63
Additional Functionality in Existing Ant Tasks . . . . .	1-63
Web Service Changed Features In This Release . . . . .	1-64
wsdl2Service Ant Task . . . . .	1-64
JMS-Implemented Web Services . . . . .	1-64
Command-Line Version of the servicegen Ant Task . . . . .	1-64

WebLogic Tuxedo Connector Features . . . . .	1-64
Enhanced Security Administration. . . . .	1-64
WLEC to WebLogic Tuxedo Connector Migration Guide. . . . .	1-65
Asynchronous tpacall . . . . .	1-65
Runtime WTC ORB . . . . .	1-65
Enhancements to FML Interface . . . . .	1-65
VIEW Buffer Example. . . . .	1-66
Changes to WTC in the Navigation Tree. . . . .	1-66
XML Features . . . . .	1-66
Developer Tools . . . . .	1-66
New appc Compiler. . . . .	1-66
New J2EE Client .JARs. . . . .	1-67
Builder Changes . . . . .	1-67
Internationalization Features. . . . .	1-67
Certifications. . . . .	1-68
Standards Support . . . . .	1-68
Deprecated Features and APIs . . . . .	1-70
Third-Party JAR Files . . . . .	1-70
Other Available Resources. . . . .	1-72
Fast Track Procedures . . . . .	1-72
Examples . . . . .	1-72
Introduction . . . . .	1-73
Additional Documentation. . . . .	1-73
Newsgroups . . . . .	1-73
Dev2Dev Online. . . . .	1-73

## 2. WebLogic Server 8.1 Known Issues

Known Issues in WebLogic Server 8.1 SP5. . . . .	2-1
--	-----

Core WebLogic Server Known Issues . . . . .	2-2
Security Known Issues . . . . .	2-2
Known Issues in WebLogic Server 8.1 SP4 . . . . .	2-2
Administration Console Known Issues . . . . .	2-4
Cluster Known Issues . . . . .	2-5
Core WebLogic Server Known Issues . . . . .	2-6
Deployment Known Issues . . . . .	2-8
Documentation Known Issues . . . . .	2-8
Examples Known Issues . . . . .	2-9
Installation Known Issues . . . . .	2-9
JDBC Known Issues . . . . .	2-10
Node Manager Known Issues . . . . .	2-10
Security Known Issues . . . . .	2-11
Transactions Known Issues . . . . .	2-12
Web Services Known Issues . . . . .	2-13
WebLogic Tuxedo Connector (WTC) Known Issues . . . . .	2-13
WebLogic Type 4 JDBC Drivers Known Issues . . . . .	2-14
Known Issues in WebLogic Server 8.1 SP3 . . . . .	2-14
Administration Console Known Issues . . . . .	2-16
Cluster Known Issues . . . . .	2-17
Core WebLogic Server Known Issues . . . . .	2-18
Documentation Known Issues . . . . .	2-21
Examples Known Issues . . . . .	2-21
Installation Known Issues . . . . .	2-21
JDBC Known Issues . . . . .	2-22
RMI/IIOP Known Issues . . . . .	2-23
Security Known Issues . . . . .	2-23
Transactions Known Issues . . . . .	2-24

Web Services Known Issues . . . . .	2-25
WebLogic Tuxedo Connector (WTC) Known Issues. . . . .	2-25
Known Issues in WebLogic Server 8.1 SP2. . . . .	2-25
Patch Available for Sun Java 2 1.4.2 SDK, Oracle 10g Driver, and SQL Server, Sybase or DB2 Database Users . . . . .	2-26
Administration Console Known Issues. . . . .	2-27
Configuration Wizard Known Issues. . . . .	2-27
Connector Known Issues . . . . .	2-28
CORE Known Issues . . . . .	2-30
EJB Known Issues. . . . .	2-31
Examples Known Issues. . . . .	2-32
General Known Issues . . . . .	2-32
Installation Known Issues . . . . .	2-33
JMS Known Issues . . . . .	2-34
JVM Known Issues . . . . .	2-35
Plug-In Known Issues. . . . .	2-36
Security Known Issues . . . . .	2-37
Servlet & JSP Known Issues . . . . .	2-39
Transactions Known Issues . . . . .	2-39
WebLogic JDBC Type 4 Drivers Known Issues. . . . .	2-40
Web Services Known Issues. . . . .	2-44
Known Issues in WebLogic Server 8.1 SP1. . . . .	2-46
Administration Console Known Issues. . . . .	2-47
Cluster Known Issues. . . . .	2-47
Connector Known Issues . . . . .	2-48
EJB Known Issues. . . . .	2-49
Examples Known Issues. . . . .	2-50
JVM Known Issues . . . . .	2-51

Miscellaneous Known Issues . . . . .	2-52
Node Manager Known Issues . . . . .	2-52
Plug-Ins Known Issues . . . . .	2-53
Security Known Issues . . . . .	2-53
Servlets & JSPs Known Issues . . . . .	2-55
System Administration Known Issues . . . . .	2-55
WebLogic JDBC Type 4 Driver for MS SQL Server Known Issues . . . . .	2-56
WebLogic Workshop Known Issues . . . . .	2-59
Web Services Known Issues . . . . .	2-59
WLEC Known Issues . . . . .	2-61
WTC Known Issues . . . . .	2-61
Known Issues in WebLogic Server 8.1 . . . . .	2-62
Administration Console Known Issues . . . . .	2-64
Classloader Known Issues . . . . .	2-66
Configuration Wizard Known Issues . . . . .	2-67
CORE Known Issues . . . . .	2-68
Deployment Known Issues . . . . .	2-69
EJB Known Issues . . . . .	2-69
Examples Known Issues . . . . .	2-72
Installation Known Issues . . . . .	2-74
Interoperability Known Issues . . . . .	2-74
JVM Interoperability . . . . .	2-75
J2EE Known Issues . . . . .	2-76
JDBC Known Issues . . . . .	2-76
JMS Known Issues . . . . .	2-79
JVM Known Issues . . . . .	2-80
Oracle Thin Driver Known Issues . . . . .	2-81
RMI/RMI-IIOP Known Issues . . . . .	2-83

Security Known Issues . . . . .	2-85
Servlets & JSPs Known Issues . . . . .	2-89
System Administration Known Issues . . . . .	2-90
Tools Known Issues . . . . .	2-91
WebLogic jDriver for Oracle Known Issues . . . . .	2-92
WebLogic Tuxedo Connector Known Issues . . . . .	2-93
WebLogic Workshop Known Issues . . . . .	2-95
Web Services Known Issues . . . . .	2-95
XML Known Issues . . . . .	2-96

### 3. Resolved Problems for Service Pack 5

Administration Console . . . . .	3-3
Cluster . . . . .	3-8
Connector . . . . .	3-8
CORBA . . . . .	3-9
Core WebLogic Server . . . . .	3-10
Deployment . . . . .	3-15
EJB . . . . .	3-15
Installer . . . . .	3-23
J2EE . . . . .	3-23
jCOM . . . . .	3-23
JDBC . . . . .	3-24
jDriver . . . . .	3-30
JMS . . . . .	3-30
JNDI . . . . .	3-35
JTA . . . . .	3-35
JVM . . . . .	3-37
Node Manager . . . . .	3-37

Operations, Administration, and Management .....	3-39
Plug-Ins .....	3-47
RMI .....	3-51
Security .....	3-53
Servlets and JSPs .....	3-57
SNMP .....	3-63
Tools .....	3-64
Utilities .....	3-65
Web Services .....	3-65
WebLogic Tuxedo Connector .....	3-70
XML .....	3-72

## 4. Resolved Problems for Service Pack 4

Administration Console .....	4-3
Classloaders .....	4-8
Clusters .....	4-8
Connector .....	4-8
Core WebLogic Server .....	4-10
Deployment .....	4-16
EJB .....	4-20
Installation .....	4-27
Internationalization .....	4-27
J2EE .....	4-27
JCOM .....	4-29
JDBC .....	4-29
jDriver .....	4-32
JMS .....	4-32
JNDI .....	4-36

JSP .....	4-36
JTA .....	4-38
JTS .....	4-39
JVM .....	4-39
Node Manager .....	4-40
Operations, Administration, and Management .....	4-41
Plug-Ins .....	4-48
RMI/RMI-IIOP .....	4-54
Security .....	4-57
Servlets .....	4-63
SNMP .....	4-69
Tools .....	4-69
Web Services .....	4-70
WebLogic Tuxedo Connector .....	4-76
WebLogic Workshop .....	4-78
XML .....	4-78

## 5. Resolved Problems for Service Pack 2

Administration Console .....	5-2
ClassLoader .....	5-6
Cluster .....	5-6
Connector .....	5-8
CORE .....	5-10
Deployment .....	5-14
EJB .....	5-17
Installation .....	5-26
J2EE .....	5-27
JCOM .....	5-28

JDBC .....	.5-28
JMS .....	.5-33
JNDI .....	.5-38
JTA .....	.5-40
Node Manager .....	.5-42
Plug-Ins .....	.5-43
RMI/RMI-IIOP .....	.5-45
Security .....	.5-49
Servlets & JSPs .....	.5-54
System Administration .....	.5-69
Tools .....	.5-75
Web Services .....	.5-78
WTC .....	.5-88
XML .....	.5-91
.....	.5-91

## 6. Resolved Problems for Service Pack 1

Administration Console .....	.6-3
ClassLoader .....	.6-8
Cluster .....	.6-9
Connector .....	.6-9
CORE .....	.6-13
Deployment .....	.6-17
EJB .....	.6-19
Installation .....	.6-28
Internationalization .....	.6-29
J2EE .....	.6-30
JDBC .....	.6-31

JMS .....	6-36
JNDI .....	6-42
JTA .....	6-43
JVM .....	6-46
Localization .....	6-46
Node Manager .....	6-47
Plug-Ins .....	6-47
RMI/RMI-IIOP .....	6-49
Security .....	6-51
Servlets & JSPs .....	6-54
System Administration .....	6-64
Tools .....	6-70
Utilities .....	6-71
WebLogic Workshop .....	6-72
Web Services .....	6-73
XML .....	6-78

## 7. Resolved Problems for Service Pack 3

Administration Console .....	7-3
Classloaders .....	7-5
Clusters .....	7-7
Compilers .....	7-10
Configuration Wizard .....	7-10
Connector .....	7-12
Core WebLogic Server .....	7-14
Deployment .....	7-22
EJB .....	7-26
Installation .....	7-32

J2EE .....	7-33
JCOM.....	7-34
JDBC.....	7-35
jDriver.....	7-41
JMS .....	7-43
JNDI .....	7-49
JSP.....	7-49
JTA.....	7-53
JVM .....	7-55
Node Manager .....	7-56
Operations, Administration, and Management .....	7-56
Plug-Ins.....	7-65
RMI/RMI-IIOP .....	7-74
Samples.....	7-77
Security.....	7-77
Servlets.....	7-86
SNMP.....	7-96
Tools .....	7-96
Web Services.....	7-98
WebLogic Type 4 JDBC Drivers .....	7-108
WebLogic Tuxedo Connector .....	7-111
WLEC .....	7-113
XML.....	7-113

# WebLogic Server 8.1 Features and Changes

Welcome to BEA WebLogic Server 8.1! As the leading application server, BEA WebLogic Server™ implements J2EE 1.3 technologies, Web services, and other leading Internet standards to provide a reliable framework for highly available, scalable, and secure applications. WebLogic Server's seamless integration of disparate, heterogeneous platforms and applications enables your network to leverage existing software investments and share the enterprise-class services and data that are crucial to building mission-critical e-business applications.

The following sections describe the new features and major improvements made in the WebLogic Server 8.1 general release and its associated service packs.

- [“What’s New in WebLogic Server 8.1 SP5” on page 1-2](#)
- [“What’s New in WebLogic Server 8.1 SP4” on page 1-6](#)
- [“What’s New in WebLogic Server 8.1 SP3” on page 1-15](#)
- [“What’s New in WebLogic Server 8.1 SP2” on page 1-27](#)
- [“What’s New in WebLogic Server 8.1 SP1” on page 1-32](#)
- [“What’s New in WebLogic Server 8.1” on page 1-37](#)
- [“Certifications” on page 1-68](#)
- [“Standards Support” on page 1-68](#)
- [“Deprecated Features and APIs” on page 1-70](#)
- [“Third-Party JAR Files” on page 1-70](#)

- [“Other Available Resources” on page 1-72](#)

## What’s New in WebLogic Server 8.1 SP5

This section details major differences between WebLogic Server 8.1 SP5 and earlier versions. It includes information about the following:

- [“Enhanced Support for Oracle RAC” on page 1-3](#)
  - [“XA Call Retry for Oracle RAC” on page 1-3](#)
  - [“Using MultiPools with Global Transactions” on page 1-3](#)
- [“Administration Console Changes” on page 1-3](#)
  - [“DISCOVERMANAGEDSERVER” on page 1-3](#)
  - [“SocketBufferSizeAsChunkSize” on page 1-4](#)
- [“Optimistic Concurrency for EJBs” on page 1-4](#)
- [“HTTP GET Request Limit” on page 1-4](#)
- [“Bypassing Compilation Errors in JSP” on page 1-4](#)
- [“J-Integra Version” on page 1-4](#)
- [“RSA Libraries” on page 1-4](#)
- [“Form-based Authentication over HTTPS” on page 1-5](#)
- [“Limiting Number of Concurrent Requests for a Session” on page 1-5](#)
- [“Option for Warning Message BEA-012034” on page 1-5](#)
- [“Accessing the Java weblogic.marathon.ddinit.EarInit Method in a Multi-User Environment” on page 1-5](#)
- [“Behavior of InitialContext” on page 1-5](#)
- [“Custom Classloader Structure with Iterative Development” on page 1-6](#)

## Enhanced Support for Oracle RAC

### XA Call Retry for Oracle RAC

Occasionally, when one RAC node fails, there may be a delay before the transaction branch on the failed node becomes available on other nodes in the RAC cluster. This prevents incomplete transactions from being properly completed, which could further result in unresolved database locks or data loss. To protect against the potential consequences of such a delay, WebLogic Server provides two configuration attributes that will enable XA call retry for Oracle RAC:

`XARetryDurationSeconds` and `XARetryIntervalSeconds`. These attributes enable the WebLogic Server transaction manager to retry XA recover, commit, and rollback calls for the specified amount of time to compensate for potential RAC failover delays. For more information, see [http://e-docs.bea.com/wls/docs81/jdbc/oracle\\_rac.html](http://e-docs.bea.com/wls/docs81/jdbc/oracle_rac.html)

### Using MultiPools with Global Transactions

You can use MultiPools to connect WebLogic Server to multiple Oracle RAC nodes. First configure a JDBC connection pool for each RAC instance in your RAC cluster with the Oracle Thin driver. Then configure a MultiPool using either the algorithm for load balancing or the algorithm for high availability and add the connection pools to the MultiPool. For information, see "[Configuration Options in WebLogic Server with Oracle RAC](#)" in Programming WebLogic JDBC.

A MultiPool "pins" a transaction to one and only one Oracle RAC instance and failover is handled at the MultiPool level when a RAC instance becomes unavailable. If there is a failure on a RAC instance before PREPARE, the operation is retried until the retry duration has expired. If there is a failure after PREPARE the transaction is failed over to another instance.

**Note:** MultiPools with global (XA) connections are supported only when used with Oracle RAC.

For more information see: [http://e-docs.bea.com/wls/docs81/jdbc/oracle\\_rac.html](http://e-docs.bea.com/wls/docs81/jdbc/oracle_rac.html)

## Administration Console Changes

### DISCOVERMANAGEDSERVER

The WebLogic Administration Console has a new option, DISCOVERMANAGEDSERVER. Selecting this option allows you to re-establish administrative control over Managed Servers after an Administration Server has already started. This option has been provided on the domain -> control page.

## SocketBufferSizeAsChunkSize

The `SocketBufferSizeAsChunkSize` attribute has been added to the Server > Configuration > Tuning tab. Enabling this attribute sets the WebLogic Server buffer size, for sending or receiving data through a raw socket, to 4KB.

When this attribute is disabled, the Operating System determines the buffer size.

**Note:** While use of this option can help improve performance on some Operating Systems, it is not guaranteed and hence, not recommended. Performance improvements are dependent on the Operating System hosting the server.

## Optimistic Concurrency for EJBs

The 'select for updates' operation is no longer used for optimistic concurrency EJBs that use Oracle drivers that implement the proprietary `sendBatch()` API. As a result, for optimistic failures, it is no longer possible to report errors at the pk level.

## HTTP GET Request Limit

Earlier, when the ISAPI plug-in for WebLogic Server encountered a HTTP GET request larger than 4096 bytes, the 'request too long: XXXX, max is 4096' error was recorded in the `wlproxy` log files. This limitation no longer exists and the ISAPI plug-in for WebLogic Server now accommodates requests of all sizes.

## Bypassing Compilation Errors in JSP

A `-k` option has been introduced in `appc` to ensure JSP compiles even after a compilation error is encountered. For `wlappc` tasks, the `continueCompilation` attribute has been added.

## J-Integra Version

WebLogic Server now includes JCOM with J-Integra 2.3.

## RSA Libraries

WebLogic Server ships with the RSA Crypto-J 3.5 library. Check the validation lists at <http://csrc.nist.gov/cryptval/> for FIPS140 compliance status.

## Form-based Authentication over HTTPS

You can now enable form-based authentication over HTTPS even if cookies are disabled in the browser. However, enabling this feature forces the browser to display the Authorization Cookie as part of the URL. To enable this feature, add the:

```
-Dweblogic.http.AuthCookieURLRewritingEnabled=true
```

flag to the WebLogic start script.

## Limiting Number of Concurrent Requests for a Session

The `weblogic.http.session.maxConcurrentRequest` property has been added to limit the number of concurrent requests for a session. If the number of concurrent requests for a given session exceeds the specified value, the servlet container will start rejecting requests. By default, this property is set to `-1`, which indicates the servlet container does not impose any restrictions.

## Option for Warning Message BEA-012034

BEA-012034 can now be disabled using the `<disable-element>` in `weblogic-ejb-jar.xml`.

## Accessing the Java `weblogic.marathon.ddinit.EarInit` Method in a Multi-User Environment

Prior to WebLogic Server 8.1 SP5, using the `java weblogic.marathon.ddinit.EarInit` method in a multi-user environment resulted in access issues for the temporary directory created by the method. And, only one user could write to the temporary directory at any given time.

You can now specify your own temporary directory using the `-Dmarathon.tmpdir` flag when you invoke the `weblogic.marathon.ddinit.EarInit` method.

## Behavior of `InitialContext`

The behavior of `InitialContext` has changed, resulting in increased memory allocation. If you create an `InitialContext` in the threads created by your application, you must subsequently close that `Context` to release resources and avoid any potential memory leaks.

See *[Programming WebLogic JNDI](#)* for more information.

## Custom Classloader Structure with Iterative Development

When a new classloader-structure element is added as a leaf node anywhere in the existing class-loader hierarchy, then the module added to the new classloader-structure can be deployed without redeploying the entire application. However, when deleting or rearranging the existing classloader-structure element within the hierarchy, the entire application should be redeployed.

When you add new module-uri(s) to an existing classloader-structure, ensure that it is added only after the existing module-uri(s). New module(s) can be deployed without redeploying the entire application. However, when moving or deleting module-uri(s) across classloader-structure elements, ensure that you redeploy the entire application.

## What's New in WebLogic Server 8.1 SP4

This section details major differences between WebLogic Server 8.1 SP4 and earlier versions. This section includes information about the following:

- [“Patch to Provide XA Support for Oracle RAC” on page 1-6](#)
- [“Patch to Support Use with Oracle 9i RAC” on page 1-9](#)
- [“WebLogic Type 4 JDBC Drivers” on page 1-11](#)
- [“EJB Features” on page 1-14](#)
- [“Security Features” on page 1-14](#)
- [“Servlet Features” on page 1-15](#)

## Patch to Provide XA Support for Oracle RAC

If you are using WebLogic Server 8.1SP4 with Oracle 9i Real Application Clusters (RAC) or Oracle 10g RAC, you should download and install the patch at [http://dev2dev.bea.com/wlsrserver/patch/wls81sp4\\_MP\\_OracleRAC\\_patch.html](http://dev2dev.bea.com/wlsrserver/patch/wls81sp4_MP_OracleRAC_patch.html)

This patch includes resolutions for issues described in Oracle's bug 3428146 and 395790. In some failure conditions, there is a window of time in which transaction branches and the data associated with them are not available across the RAC cluster, which prevents incomplete transactions from being properly completed, which may result in database deadlocks or data loss. To work around these issues, BEA added the `XA_RETRY_DURATION_SECONDS` attribute to the JDBC connection pool configuration. This attribute enables the WebLogic Server transaction manager to retry XA recover,

commit, and rollback calls for the specified amount of time to compensate for potential RAC failover delays.

In addition, this patch supports the use of WebLogic JDBC MultiPools for applications using global transactions (XA). MultiPools offer database connection failover without the limitations and known issues associated with a connection pool configuration using connect-time failover. MultiPools also offer load balancing for XA, and additional features such as callbacks and automatic failback.

**Note:** If you have already installed the Oracle 9i RAC patch (wlplat81sp4\_Oracle9iRAC\_patch.jsp) BEA recommends replacing that patch with this one regardless whether you plan to use MultiPools. This patch contains all the fixes included in the Oracle 9i RAC patch and will be the patch updated for WLS if future updates are needed.

## Patch Configuration Requirements

After installing this patch, you must update your domain configuration. For more information about updating your domain configuration, see [“Enable XA Call Retry for Oracle RAC” on page 1-7](#).

If you are using WebLogic Server 8.1SP3 with Oracle 9i RAC, BEA recommends that you upgrade to WebLogic Server 8.1SP4 and apply the wlplat81sp4\_Oracle10gRAC\_patch.

See [“XA Considerations and Limitations with Oracle 9i RAC”](#) in *Programming WebLogic JDBC* for configuration details and known limitations when using WebLogic Server with Oracle 9i RAC.

For more information about Oracle RAC limitations, contact Oracle.

For the latest information on the hardware and software platforms supported by WebLogic Server and BEA WebLogic Platform™, see [Supported Configurations](#).

## JDBC MultiPool Configuration for WebLogic Server with Oracle RAC

To use MultiPools to connect WebLogic Server to multiple Oracle RAC nodes, first configure a JDBC connection pool for each RAC instance in your RAC cluster with the Oracle Thin driver. Then configure a MultiPool using either the algorithm for load balancing or the algorithm for high availability and add the connection pools to the MultiPool. You can use the Administration Console or any other means that you prefer to configure your domain, such as the weblogic.Admin command line utility, the Weblogic Scripting Tool (WLST), or a JMX program. For information, see [“Configuration Options in WebLogic Server with Oracle RAC”](#) in *Programming WebLogic JDBC*.

## Enable XA Call Retry for Oracle RAC

Occasionally, when one RAC node fails over to another, there may be a delay before the transaction branches on the now failed node, and the data associated with them, are available throughout the

cluster. This prevents incomplete transactions from being properly completed, which could further result in unresolved database locks or data loss. To protect against the potential consequences of such a delay, WebLogic Server provides two configuration attributes that will enable XA call retry for Oracle RAC: `XARetryDurationSeconds` and `XARetryIntervalSeconds`.

To enable XA call retries, add a value for the `XARetryDurationSeconds` attribute to all JDBC connection pools in your WebLogic domain that connect to an Oracle RAC instance. For example:

```
<JDBCConnectionPool
    Name="oracleRACPool "
    DriverName="oracle.jdbc.xa.client.OracleXADataSource"
    ...
    XARetryDurationSeconds=480
/>
```

**Note:** The `XARetryDurationSeconds` attribute is not available in the Administration Console. To enable this feature, you must manually edit your `config.xml` file or change the configuration using the `weblogic.Admin` command line utility or a JMX program.

Use the following formula to determine the value for `XARetryDurationSeconds`:

`XARetryDurationSeconds` = (longest transaction timeout for transactions that use connections from the connection pool) + (delay before XIDs are available on all RAC nodes, typically less than 5 minutes)

For example, if your application sets the longest transaction timeout as 180 seconds, you should set `XARetryDurationSeconds` to 180 seconds + 300 seconds, for a total of 480 seconds.

**Note:** It is generally better to set `XARetryDurationSeconds` higher than minimally necessary to make sure that all transactions are completed properly. Setting the value higher than minimally required should not affect application performance during normal operations. The additional processing only affects transactions that have been prepared but have failed to complete.

You can also optionally set a value for the `XARetryIntervalSeconds` attribute. This value determines the time between XA retry calls. By default, the value is 60 seconds. Decreasing the value will decrease the amount of time between XA retry attempts. The default value should suffice in most cases.

## Patch to Support Use with Oracle 9i RAC

If you are using WebLogic Server 8.1SP4 with Oracle 9i Real Application Clusters (RAC), you should download and install the patch at

[http://commerce.bea.com/d2d/wlplat81sp4\\_Oracle9iRAC\\_patch.jsp](http://commerce.bea.com/d2d/wlplat81sp4_Oracle9iRAC_patch.jsp)

This patch includes resolutions for issues described in Oracle's bug 3428146 and 395790. In some failure conditions, there is a window of time in which transaction IDs are not available across the RAC cluster, which prevents incomplete transactions from being properly completed, which further results in deadlocks in the database. To work around these issues, BEA added the

`XARetryDurationSeconds` attribute to the JDBC connection pool configuration. This attribute enables the WebLogic Server transaction manager to retry XA recover, commit, and rollback calls for the specified amount of time to compensate for potential RAC failover delays.

Configuration for the `wlplat81sp4_Oracle9iRAC_patch` described here is in addition to other configuration requirements in WebLogic Server and in the Oracle RAC cluster. For more details about configuring WebLogic Server to connect to an Oracle RAC database, see "Using WebLogic Server with Oracle RAC" in Programming WebLogic JDBC.

### Patch Configuration Requirements

After installing this patch, you must update your domain configuration. See the following sections for more details:

- "Enable XA Call Retry" on page 1-9
- "Target Connection Pools to All Servers" on page 1-10

If you are using WebLogic Server 8.1SP3 with Oracle 9i RAC, BEA recommends that you upgrade to WebLogic Server 8.1SP4 and apply the `wlplat81sp4_Oracle9iRAC_patch`.

See "XA Considerations and Limitations with Oracle RAC" in *Programming WebLogic JDBC* for configuration details and known limitations when using WebLogic Server with Oracle RAC.

For more information about Oracle RAC limitations, contact Oracle.

For the latest information on the hardware and software platforms supported by WebLogic Server and WebLogic Platform, see *Supported Configurations*.

### Enable XA Call Retry

To enable XA call retries, add a value for the `XARetryDurationSeconds` attribute to all JDBC connection pools in your WebLogic domain that connect to an Oracle RAC instance. For example:

```
<JDBCConnectionPool
  Name="oracleRACPool"
  DriverName="oracle.jdbc.xa.client.OracleXADataSource"
  ...
  XARetryDurationSeconds=480
/>
```

**Note:** The `XARetryDurationSeconds` attribute is not available in the Administration Console. To enable this feature, you must manually edit your `config.xml` file or change the configuration using the `weblogic.Admin` command line utility or a JMX program.

Use the following formula to determine the value for `XARetryDurationSeconds`:

`XARetryDurationSeconds` = (longest transaction timeout for transactions that use connections from the connection pool) + (delay before XIDs are available on all RAC nodes, typically less than 5 minutes)

For example, if your application sets the longest transaction timeout as 180 seconds, you should set `XARetryDurationSeconds` to 180 seconds + 300 seconds, for a total of 480 seconds.

**Note:** It is generally better to set `XARetryDurationSeconds` higher than minimally necessary to make sure that all transactions are completed properly. Setting the value higher than minimally required should not affect application performance during normal operations. The additional processing only affects transactions that have been prepared but have failed to complete.

You can also optionally set a value for the `XARetryIntervalSeconds` attribute. This value determines the time between XA retry calls. By default, the value is 60 seconds. Decreasing the value will decrease the amount of time between XA retry attempts. The default value should suffice in most cases.

## Target Connection Pools to All Servers

With this patch, all JDBC connection pool/data source pairs that connect to an Oracle RAC instance must be targeted to all WebLogic Server instances that participate in global transactions for which RAC is a participant.

During server startup, if a resource checkpoint is read from the transaction log, and the resource is found to be available on another server, then it is possible that recovery for that resource can complete before the resource is registered locally. If the resource is configured without the `XARetryDurationSeconds` attribute, recovery for the resource will complete without XA call retries, which can result in deadlocked data. By targeting the connection pool to all relevant WebLogic Server instances, XA call retries will be initiated on any server that performs the recovery.

For instructions about targeting a data source and connection pool, see the following sections in the *Administration Console Online Help*:

- [“Deploying a Connection Pool to One or More Servers or Clusters”](#)
- [“Deploying a JDBC Data Source to a Server or Cluster”](#)

## WebLogic Type 4 JDBC Drivers

In WebLogic Server 8.1SP4, the WebLogic Type 4 JDBC drivers were updated. The drivers now include the following new enhancements and features. For more information, see [WebLogic Type 4 JDBC Drivers](#).

### All Drivers

The following enhancements were made to all WebLogic Type 4 JDBC drivers:

- Client-side load balancing
- More flexible connection failover
- Connection retry
- Ability to set and retrieve client information on a connection
- Parameter metadata support for simple inserts and updates
- ResultSet metadata support has been enhanced to return valid table name information:
  - The drivers return the name of the table to which the column belongs from the `ResultSetMetaData.getTableName()` method for columns in a result set that map to a column in a table in the database. The drivers return an empty string from the `ResultSetMetaData.getTableName()` method for columns in a result set that do not map to a column in a table (for example, aggregates, literals, and so forth).
  - The SELECT statements for which ResultSet metadata is returned may contain aliases, joins, and fully qualified names.
  - The drivers return schema name and catalog name information when the driver can determine that information.

To provide this enhanced functionality, the driver must perform additional processing. To ensure that applications that do not require table information do not have to pay a performance penalty for this additional processing, a new integer connection option,

`ResultSetMetaDataOptions`, has been added to control whether the driver returns valid table name information. Valid values are 1 and 0. The default is 0.

- Parameter metadata support has been enhanced to return parameter metadata for SELECT statements. When connected to a database that does not support returning parameter metadata for SELECT statements natively, the drivers do support returning parameter metadata for many common forms of ANSI SQL SELECT statements. Database-specific SELECT syntax is not currently supported.

## DB2 Driver

At the time of the WebLogic Server 8.1SP4 release, WebLogic Server was not certified for use with DB2. The DB2 driver was not included in the WebLogic Server 8.1SP4 distribution. A patch is now available for WebLogic Server Service Pack 4 at the following URL:

[http://dev2dev.bea.com/products/wlplatform81/patch/wlplat81sp4\\_db2\\_patch.jsp](http://dev2dev.bea.com/products/wlplatform81/patch/wlplat81sp4_db2_patch.jsp)

## Informix Driver

- A new connection option, `DBDate`, was added to set the value of the Informix `DBDate` server environment variable. This value controls how the driver interprets literal date values when inserting or updating data in DATE columns and how it formats date strings retrieved from DATE columns. Using this option, you can customize:
  - The Order in which the month, day, and year fields appear in a date string
  - The Year field to contain two or four digits
  - The Separator character used to separate the date fields

## SQL Server Driver

- Improved trigger result and error handling
- A new connection option, `ReceiveStringParameterType`, was added to allow you to specify how the driver describes String output parameters to the database. Valid values are NVARCHAR, VARCHAR, and DESCRIBE. The default is NVARCHAR.

## Oracle Driver

- New Oracle10g support, including support for the Oracle10g BINARY\_FLOAT and BINARY\_DOUBLE data types
- Improved JTA performance

- Ability to retrieve connection information, including client-side load balancing information from an Oracle `tnsnames.ora` file
- Ability to set Dedicated or Shared server type
- Driver behavior has been changed for the connection option `CatalogOptions`, which determines the type of information included in result sets returned from catalog functions. Behavior has been changed as follows:
  - If set to 0, result sets contain default `DatabaseMetaData` results.
  - If set to 1, result sets contain remarks information returned from the `DatabaseMetaData` method `getTables`. Result sets contain remarks and column default information returned from the `DatabaseMetaData` method `getColumns`.
  - If set to 2, result sets contain synonyms returned from the `DatabaseMetaData` methods: `getColumns`, `getProcedures`, `getProcedureColumns`, and `getIndexInfo`.
  - If set to 3, result sets contain remarks, column defaults, and synonyms (as described in options 1 and 2).

The default value is 2.

- A new connection option was introduced, `SendFloatParametersAsString`, to specify whether float and double parameters are sent to the server as a string or as a floating point number. Valid values are true and false. When `SendFloatParametersAsString` is false, the driver sends float and double parameters as floating point numbers. When `SendFloatParametersAsString` is set to true, the driver sends float and double parameters to the database server as string values. Setting `SendFloatParametersAsString` to true provides backwards compatibility with previous driver versions. The default value is false.
- A new connection option was introduced, `CodePageOverride`, to control the way the driver performs character conversions when communicating with an Oracle server. When set to a valid value, the specified code page overrides the code page used by the driver to convert character data to the database character set. This option has no effect on how the driver converts character data to the national character set. If `CodePageOverride` is not specified, the driver automatically determines the code page used to convert character data based on the Oracle server's reported database character set.

## Sybase Driver

- A new Boolean connection option, `UseAlternateProductInfo`, was introduced to allow the user to specify that the driver return more accurate information for the `DatabaseMetaData.getDatabaseProductName()` and `DatabaseMetaData.getDatabaseProductVersion()` methods. When set to false, the

driver returns the information it receives from the server during the login process. This is the version information that the driver has always returned. When set to true, the driver makes an additional query and returns the information it receives from selecting the value of @@version. The default value is false.

## EJB Features

### Dynamic Finders

Dynamic Finders can now use the relationship-caching feature. The dynamic finders need to be constructed as follows:

```
Query query = qh.createQuery();
    Properties props = new Properties();
    props.setProperty("GROUP_NAME", "byNameGroup");
    props.setProperty("SQL_SELECT_DISTINCT", "true");
    props.setProperty("RELATIONSHIP_CACHING_NAME", "cacheMoreBeans");
    Collection results = query.find(ejbql, props);
```

You can set both field-group as and relationship-caching on the dynamic query.

### PreparedQuery Interface

The PreparedQuery public interface that extends the QueryProperties interface has been added to the weblogic.ejb package. For more information, see <http://e-docs.bea.com/wls/docs81/javadocs/weblogic/ejb/PreparedQuery.html>

## Security Features

### Trust Manager

WebLogic Server 8.1 now has a property that can override the default TrustManager implementation. When this property is set, the custom TrustManager is called instead of the default WebLogic Server TrustManager.

The custom TrustManager may behave differently and cause certificate validation to fail or succeed for different reasons. The custom TrustManager is used for incoming connections and not for outgoing connections. It should be tested with incoming connections.

Future releases of WebLogic Server may provide a different implementation for custom TrustManager functionality.

## LDAP

WebLogic Server has a new property, `(servername).membership.directmembershiponly`, that can be specified when configuring the LDAP realm v2. The `(servername)` can be the name of any server or can be left blank.

If this property is set to true, the LDAP search filter will only list the members of a group who are direct members to the group. For example, if User X belongs to Group G1 and Group G1 belongs to Group G2, only Group G1 will be listed if this property is set to true. By default, this property is set to false.

## Encryption Utility

There is a new `weblogic.security.Encrypt` utility that encrypts cleartext strings for use with WebLogic Server. The utility uses the encryption service of the current directory, or the encryption service for a specified WebLogic Server domain root directory.

## Single Pass Identity Assertion

Single Pass Identity Assertion provider allows single sign-on with desktop clients.

## Servlet Features

`WebAppServletContext.getResourcePaths()` now returns both directories and files under the web-app root directory as well as any virtual directories configured in the `weblogic.xml` for the web-app.

# What's New in WebLogic Server 8.1 SP3

This section details major differences between WebLogic Server 8.1 SP3 and earlier versions. This section includes information about the following:

- [“New SDKs Bundled with WebLogic Platform 8.1” on page 1-16](#)
- [“WebLogic Server Process Edition” on page 1-16](#)
- [“Web Services Features” on page 1-17](#)
- [“JDBC Features” on page 1-19](#)
- [“WebDAV Support” on page 1-24](#)
- [“Security Features” on page 1-25](#)
- [“WAPEnabled Attribute for HTTP Sessions” on page 1-26](#)

- [“WebLogic Tuxedo Connector \(WTC\) Features” on page 1-26](#)

## New SDKs Bundled with WebLogic Platform 8.1

The WebLogic Platform 8.1 SP3 installation includes new versions of the Java 2 SDKs:

- Sun Java 2 SDK 1.4.2\_04
- BEA JRockit® SDK 1.4.2\_04

If you are upgrading a WebLogic domain from 8.1 SP2 to 8.1 SP3, You need to modify the domain startup script to point to the location in which the new SDK was installed. This script is located in the domain's root directory. Depending on the type of domain you are upgrading, this script by default is named either `setDomainEnv` or `startWebLogic`.

To modify this script, update the value of the `JAVA_HOME` variable. For example:

```
set JAVA_HOME=C:\bea\jrockit81sp3_142_04
```

It is also recommended that you update your BEA WebLogic Workshop® applications, application startup scripts, and silent configuration scripts to reference the new Sun or JRockit SDK directory. For information about updating your Weblogic Workshop applications to use the new SDK, see the [WebLogic Platform Upgrade FAQ](#).

For more information about the changes between Sun JVM 1.4.1 and 1.4.2, see Enhancements and Changes in Java 2 SDK v1.4.2 at the Sun Microsystems Web site, at the following URL:  
<http://java.sun.com/j2se/1.4.2/changes.html>

## WebLogic Server Process Edition

WebLogic Server Process Edition is a new product offering that provides a unified solution for developing process-based applications that run in the WebLogic Server environment. You can use WebLogic Server Process Edition to build and execute process-based applications without having to purchase a full BEA WebLogic Integration™ license.

WebLogic Server Process Edition includes the following components:

- **WebLogic Server Premium Edition**—Enables full support for WebLogic Server functionality, including the core Java 2 Enterprise Edition (J2EE) features, WebLogic Workshop, and premium clustering, caching, and messaging capabilities.
- **The Business Process Management (BPM) functionality of WebLogic Integration**—Allows you to model and execute business processes that span multiple internal systems, external resources, and users, including trading partners. These business processes can orchestrate the execution

of business logic and the exchange of business documents among such systems and users in a loosely coupled fashion.

For more information, see the [WebLogic Server Process Edition documentation](#).

## Web Services Features

WebLogic Server 8.1 Service Pack 3 includes the following new and changed Web Service features.

### Implementation of the Final 1.0 OASIS Web Services Security Specification

WebLogic Web Services now implement the following [OASIS Standard 1.0 Web Services Security](#) specifications, dated April 6, 2004:

- Web Services Security: SOAP Message Security
- Web Services Security: Username Token Profile
- Web Services Security: X.509 Token Profile

WebLogic Server Service Pack 2 implemented a Working Draft Version 1.0 of the specification.

The final implementation of the Web Services Security specification is *not* compatible with the implementations in Service Pack 2 and previous versions. This means, for example, that a client application that uses the Service Pack 2 (or previous) WebLogic Web Services security APIs to digitally sign or encrypt a SOAP message cannot invoke a message-level secured Web Service deployed on a Service Pack 3 WebLogic Server instance. The reverse is also true: a client application using the Service Pack 3 WebLogic Web Services security APIs cannot invoke a secure Web Service deployed on a Service Pack 2 or previous WebLogic Server instance.

This lack of compatibility results from changes between the two versions of the Web Services Security specification that were implemented in Service Packs 2 and 3. In particular, the working draft specification of April 2002 specified that certain attributes of the security header in a SOAP message use QName values. The final April 2004 version of the specification, however, specified that those attributes use URI values instead. This change makes SOAP security consistent with other security specifications, such as XML Digital Signatures, XML Encryption, and SAML.

See [Configuring Message-Level Security \(Digital Signatures and Encryption\)](#) at [http://e-docs.bea.com/wls/docs81/webserv/security.html#message\\_level\\_security](http://e-docs.bea.com/wls/docs81/webserv/security.html#message_level_security) for details.

## HTTPBindingInfo API For SSL Socket Sharing

You can now use the `weblogic.webservice.binding.https.HttpsBindingInfo` SSL binding API, in addition to system properties, to programmatically enable socket sharing from within your SSL Web Services client application.

See [Using SSL Socket Sharing When Using the WebLogic SSL Implementation at `http://e-docs.bea.com/wls/docs81/webserv/security.html#ssl\_socket\_sharing`](http://e-docs.bea.com/wls/docs81/webserv/security.html#ssl_socket_sharing) for details.

## New wsdl2Service Attribute to Generate a Skeleton Implementation Class

The `wsdl2Service` Ant task has a new attribute (`generateImpl`) that specifies, when set to `True`, that the Ant task should generate an empty implementation class, in addition to a Java interface that represents the implementation of your Web Service based on the WSDL file.

See [wsdl2Service](http://e-docs.bea.com/wls/docs81/webserv/anttasks.html#wsdl2Service) at `http://e-docs.bea.com/wls/docs81/webserv/anttasks.html#wsdl2Service` for details.

## New servicegen and source2wsdd Attribute to Ignore Authorization Header in SOAP Request

The `servicegen` and `source2wsdd` Ant tasks have a new attribute (`ignoreAuthHeader`) that specifies, when set to `True`, that the Web Service should ignore the `Authorization` HTTP header in the SOAP request.

See [servicegen](http://e-docs.bea.com/wls/docs81/webserv/anttasks.html#servicegen) at `http://e-docs.bea.com/wls/docs81/webserv/anttasks.html#servicegen` and [source2wsdd](http://e-docs.bea.com/wls/docs81/webserv/anttasks.html#source2wsdd) at `http://e-docs.bea.com/wls/docs81/webserv/anttasks.html#source2wsdd` for details.

## WebLogic XML Digital Signatures API

The WebLogic XML Digital Signature API contains classes to digitally sign and validate SOAP messages.

See [Using the WebLogic XML Digital Signatures API at `http://e-docs.bea.com/wls/docs81/xml/xml\_xpath.html#xml\_dig\_sig\_api`](http://e-docs.bea.com/wls/docs81/xml/xml_xpath.html#xml_dig_sig_api) for more details.

## WebLogic Web Service Security API to Add a Timestamp to a SOAP Message

The `weblogic.xml.security.wsse.Security` API has a new method, `addTimestamp()`, which you can use when invoking a secure non-WebLogic Web Service to add a timestamp, and optional expiration date, to the security element of the SOAP request.

For more information, see the updated example in [Writing the Java Code to Invoke a Secure Non-WebLogic Web Service](#) at [http://e-docs.bea.com/wls/docs81/webserv/security.html#invoke\\_secure\\_non\\_wls\\_service](http://e-docs.bea.com/wls/docs81/webserv/security.html#invoke_secure_non_wls_service).

## Change in clientgen Behavior Due to Stricter Compliance With WS-I

The `clientgen` Ant task now strictly enforces the required sections of the [Web Services Interoperability \(WS-I\) Basic Profile Version 1.0 specification](#), resulting in changed behavior from previous versions of WebLogic Web Services.

For example, section 5.6.16 of the WS-I specification states that the `<fault>` element in a WSDL file must include the `name` attribute. Previously, the `clientgen` Ant task would successfully complete when run against a WSDL file that did not comply with this section of the specification; in 8.1, however, the Ant task fails with the following error:

```
[clientgen] Generating client jar for myWSDL.wsdl ...
[clientgen] weblogic.webservice.tools.wsdlp.WSDLParseException: ERROR[WSDL
Parser]:There is no name
attribute found for soap:fault in binding operation <operation
name="myOperation">
```

A consequence of this stricter compliance is that `clientgen` may now fail when run against the automatically-generated WSDL of a 7.0 WebLogic Web Service. This is because 7.0 WebLogic Web Services did not strictly comply with WS-I, and the generated WSDL did not include the `name` attribute in the `<fault>` element.

The workaround to this problem is to either manually update the WSDL to comply with WS-I, or to use [portable stubs](#).

## JDBC Features

WebLogic Server 8.1 Service Pack 3 includes the following new JDBC features and changes.

### WebLogic Server 8.1SP3 Certified with Oracle 9i RAC on HP-UX

WebLogic Server 8.1SP3 is certified for use with Oracle 9i RAC on HP-UX 11 and 11i. See "[Using WebLogic Server with Oracle RAC](#)" in *Programming WebLogic JDBC* for configuration details and requirements. Also see [Supported Configurations](#) for details and updates about certified versions.

## Deprecation of WebLogic jDriver for Oracle

With the WebLogic Server 8.1 SP3 release, the BEA WebLogic JDriver™ for Oracle is deprecated. The driver will be removed in a future release. BEA recommends that you use a different WebLogic JDBC driver to connect to your Oracle database. For information about supported database versions and drivers, see [“Supported Database Configurations”](#) in *Supported Configurations for WebLogic Platform 8.1*.

## Oracle 10g JDBC Thin Driver

In WebLogic Server 8.1 SP3, the Oracle 10g (10.1.0.2.0) version of the Oracle Thin driver was added to the release and is now the default version of the Oracle Thin driver. For more details about using the Oracle Thin driver with WebLogic Server, [“Using Third-Party Drivers with WebLogic Server”](#) in *Programming WebLogic JDBC*.

For Globalization Support with this version of the driver, Oracle supplies the `orai18n.jar` file, which replaces `nls_charset.zip`. If you use character sets other than US7ASCII, WE8DEC, WE8ISO8859P1 and UTF8 with CHAR and NCHAR data in Oracle object types and collections, you must include `orai18n.jar` in your CLASSPATH. `orai18n.jar` is *not* installed with WebLogic Server. You can download it from the Oracle Web site.

**Note:** The Oracle 10g Thin driver has a batch size limit of 1682 operations. In previous versions of the Oracle Thin driver, there was no apparent size limit. If your application runs very large batch operations, you may see batch failures. To work around this issue, limit your batch size to 1682 or use the 9.2.0 version of the Oracle Thin driver. Oracle recommends batch sizes of between 5 and 30.

## Updated WebLogic Type 4 JDBC Drivers

WebLogic Server 8.1 SP3 includes updates to the WebLogic Type 4 JDBC drivers. The updated drivers resolve some important issues.

**Note:** If you use the XA WebLogic Type 4 MS SQL Server JDBC Driver in global transactions (through JTA), you must reinstall the Microsoft SQL Server JDBC XA procedures after installing the Weblogic Server service pack. See [“Installing Stored Procedures for JTA”](#) in *BEA WebLogic Type 4 JDBC Drivers*.

## Multiple JNDI Names for Data Sources

In WebLogic Server 8.1SP3 and later releases, you can configure a data source so that it binds to the JNDI tree with multiple names. You can use a multi-named data source in place of configuring multiple data sources that point to a single JDBC connection pool.

For more details, see [“Binding a Data Source to the JNDI Tree with Multiple Names”](#) in the Administration Console Online Help.

## PinnedToThread Connection Pool Property to Increase Performance

To minimize the time it takes for an application to reserve a database connection from a connection pool and to eliminate contention between threads for a database connection, you can add the `PinnedToThread` property in the connection Properties list for the connection pool, and set its value to `true`.

When `PinnedToThread` is enabled, WebLogic Server pins a database connection from the connection pool to an execution thread the first time an application uses the thread to reserve a connection. When the application finishes using the connection and calls `connection.close()`, which otherwise returns the connection to the connection pool, WebLogic Server keeps the connection with the execute thread and does not return it to the connection pool. When an application subsequently requests a connection using the same execute thread, WebLogic Server provides the connection already reserved by the thread. There is no locking contention on the connection pool that occurs when multiple threads attempt to reserve a connection at the same time and there is no contention for threads that attempt to reserve the same connection from a limited number of database connections.

For more details, see [“Increasing Performance with the PinnedToThread JDBC Connection Pool Property”](#) in the *Administration Console Online Help*.

## JDBC MultiPool Failover Enhancements

In WebLogic Server 8.1SP3, the following enhancements were made to JDBC MultiPools:

- Connection request routing enhancements to avoid requesting a connection from a disabled connection pool within a MultiPool.
- Automatic fallback on recovery of a failed connection pool within a MultiPool.
- Failover for busy connection pools within a MultiPools with the High Availability algorithm.
- Failover callbacks for MultiPools with the High Availability algorithm.
- Fallback callbacks for MultiPools with either algorithm.

See [“MultiPool Failover Enhancements”](#) in *Programming WebLogic JDBC* for more details.

## JDBC Connection Pool Testing Enhancements

In WebLogic Server 8.1SP3, the following features were added to JDBC connection pools to improve the functionality of database connection testing for pooled connections and to minimize delays in connection request handling:

- `CountOfTestFailuresTillFlush`—Closes all connections in the connection pool after the number of test failures that you specify to minimize the delay caused by further database testing.
- `CountOfRefreshFailuresTillDisable`—Disables the connection pool after the number of test failures that you specify to minimize the delay in handling connection requests after a database failure.

To enable these features, you add the attributes to the `JDBCConnectionPool` object in the `config.xml` file. Both attributes also require that `TestConnectionsOnReserve` is set to `true` and that a value is provided for `TestTableName`. See “[JDBC Connection Pool Testing Enhancements](#)” in *Programming WebLogic JDBC* for more information.

## Connection Profiling for Troubleshooting Transaction Remnants on a JDBC Connection

In WebLogic Server 8.1SP3, the `ConnProfileEnabled` attribute was added to the JDBC connection pool configuration. When set to `true`, WebLogic Server stores the stack trace whenever a connection is released back into the connection pool. If an exception is thrown during a subsequent operation on the connection related to global (XA) transactions, WebLogic Server reports this stack trace with the exception.

You can use this feature to detect local transaction work left incomplete by application code, which can interfere with subsequent global (XA) transaction operations on the JDBC connection.

This feature uses more resources than normal connection pool operations and will likely degrade connection pool performance, so it is not recommended for production use. Also, this feature does not apply to connections created with a non-XA JDBC driver.

You can configure this attribute with the `Enable Connection Profiling` attribute on the [JDBC Connection Pool -> Configuration -> Connections](#) tab in the Administration Console or directly in the `config.xml` file. For example:

```
<JDBCConnectionPool
  ConnProfilingEnabled="true"
  DriverName="weblogic.jdbcx.oracle.OracleDataSource"
  Name="MyJDBC Connection Pool"
```

```

    Password="{3DES}GNkMrpAbFqBKQp7N8JXy+/NE0qhBW+SU"
    Properties="user=scott;portNumber=1521;SID=demo;serverName=dbserver1"
    URL="jdbc:bea:oracle://dbserver1:1521"
  />

```

**Note:** Line breaks added for readability.

## Support for XAResource Transaction Timeout

The WebLogic Server Transaction Manager now supports setting a transaction branch timeout value on a participating XA resource if the resource manager supports the

`javax.transaction.xa.XAResource.setTimeout()` method. You may want to set a transaction branch timeout if you have long-running transactions that exceed the default timeout value on the XA resource.

For the WebLogic Server Transaction Manager to set the transaction timeout on a JDBC XA resource, specify a value for the following properties in the JDBC connection pool tag in the `config.xml` file:

- `XASetTransactionTimeout`—A boolean property. When set to true, the WebLogic Server Transaction Manager calls `XAResource.setTimeout()` before calling `XAResource.start`, and passes either the `XATransactionTimeout` or the global transaction timeout in seconds. When set to false, the Transaction Manager does not call `setTimeout()`. The default value is false.
- `XATransactionTimeout`—The number of seconds to pass as the transaction timeout value in the `XAResource.setTimeout()` method. When this property is set to 0, the WebLogic Server Transaction Manager passes the global WebLogic Server transaction timeout in seconds in the method (see “JTA” in the *Administration Console Online Help*). The default value for this parameter is 0. If set, this value should be greater than or equal to the global Weblogic Server transaction timeout.

These properties apply to connection pools that use an XA JDBC driver to create database connections only. They are ignored if a non-XA JDBC driver is used.

When these values are set, the WebLogic Server Transaction Manager calls `XAResource.setTimeout()` as described above. The implementation of the method in the XA resource manager (for example, an XA JDBC driver) or the XA resource determines how the value is used. For example, for Oracle, the `setTimeout()` method sets the Session Timeout (`SesTm`), which acts as a maximum idle time for a transaction. The behavior may be different for other XA Resources.

The `XASetTransactionTimeout` and `XATransactionTimeout` properties are not available in the Administration Console. You must add them to the `config.xml` file while the domain is not active. For example:

```
<JDBCConnectionPool
  DriverName="oracle.jdbc.xa.client.OracleXADataSource"
  Name="oraclePool"
  Password="{3DES}8YdvP4FQW3k="
  Properties="user=SCOTT"
  URL="jdbc:oracle:thin:@server:port:sid"
  XASetTransactionTimeout="true"
  XATransactionTimeout="120" />
```

## SQL Statement Timeout Enhancements for Pooled JDBC Connections

In WebLogic Server 8.1SP3, the following attributes were added to JDBC connection pools to enable you to limit the amount of time that a statement can execute on a database connection from a JDBC connection pool:

- `StatementTimeout`—The time in seconds after which a statement executing on a pooled JDBC connection times out. When set to -1, (the default) statements do not time out.
- `TestStatementTimeout`—The time in seconds after which a statement executing on a pooled JDBC connection for connection initialization or testing times out. When set to -1, (the default) statements do not time out.

For more information, see [“SQL Statement Timeout Enhancements for Pooled JDBC Connections”](#) in *Programming WebLogic JDBC*.

**Note:** These attributes are not available in the Administration Console. You must manually edit the `config.xml` file to enable these features.

## WebDAV Support

WebLogic Server proxy plug-ins restrict the HTTP commands that can be submitted from the client to the server. The validation rules in the plug-in code now allow the following HTTP commands that are needed for WebDAV implementations:

```
DELETE
GET
HEAD
OPTIONS
```

POST  
PUT  
\*COPY  
LOCK  
MKCOL  
MOVE  
PROPFIND  
PROPPATCH  
SEARCH  
UNLOCK

## Security Features

In WebLogic Server 8.1 SP3, the following features were added to security:

- An LDAP X509 Identity Assertion provider which receives an X509 certificate; looks up the LDAP object for the user associated with that certificate; ensures that the certificate in the LDAP object matches the presented certificate; and then retrieves the name of the user from the LDAP object.
- A set of attributes which can be used to optimize the performance of the WebLogic and LDAP Authentication providers. The attributes can be used to improve performance in the following ways:
  - Configure the Active Directory Authentication provider to perform group membership lookups using the `tokenGroups` attribute. The `tokenGroups` attribute holds the entire flattened group membership for a user as an array of SID values. The SID values are specially indexed in the Active Directory and yield fast lookup response.
  - Optimize the configuration the group membership caches used by the WebLogic and LDAP Authentication providers.
  - Expose the internal `PrincipalValidator` cache and increase its thresholds.
- Support for hyphens in user and group names.
- Support for auditing all management operations. In WebLogic Server 8.1 SP3, the WebLogic Server security infrastructure audits all security decisions and enables you to generate server log messages or Audit Events, which can be handled by any Auditing provider configured for the default security realm.

## WAPEnabled Attribute for HTTP Sessions

The `WAPEnabled` attribute in the Administration Console (~~Servers~~—~~servername~~—~~Protocols~~—~~HTTP~~—~~Advanced Options~~—~~WAP Enabled~~) restricts the size and format of a session ID so session tracking can be used with WAP devices. It now works correctly when defining the HTTP settings for the WebLogic Server instance; in previous versions, checking the box had no affect.

See [URL Rewriting and Wireless Access Protocol \(WAP\)](http://e-docs.bea.com/wls/docs81/webapp/sessions.html#wap) at <http://e-docs.bea.com/wls/docs81/webapp/sessions.html#wap>.

## WebLogic Tuxedo Connector (WTC) Features

In BEA WebLogic Tuxedo Connector™ 8.1 SP3, the following features were added:

- XATMI buffer support

WebLogic Tuxedo Connector now supports the XATMI buffer types `X_COMMON`, `X_C_TYPE`, and `X_OCTET`.

WTC has added the following new classes for XATMI support:

- `TypedXOctet` class to support `X_OCTET`
- Abstract classes `TypedXCType` and `TypedXCommon` support `X_C_TYPE` and `X_COMMON`

Two new options in the `viewj/viewj32` compilers have been added:

- `-xcommon` creates a class that extends `TypedXCommon`.
- `-xctype` creates a class that extends `TypedXCType`

For more information, see “[WebLogic Tuxedo Connector TypedBuffers](#)” in *WebLogic Tuxedo Connector Programmer’s Guide*.

- Packed decimal support

The `FLD_DECIMAL` type in a `VIEW` buffer is supported in both WTC and the Tuxedo Control.

WebLogic Tuxedo Connector support for `FLD_DECIMAL` results in three enhancements:

- The creation of a new Java type to represent the Tuxedo packed decimal type. This type is called `weblogic.wtc.jatmi.Decimal` and is an extension of the `java.lang.Number` class.
- Enhancements to the `viewj/viewj32` compilers to generate both a single packed decimal field value and an array of such values.

## What's New in WebLogic Server 8.1 SP2

This section details major differences between WebLogic Server 8.1 SP2 and earlier versions. This section includes information about the following:

- “Patch Available for Sun Java 2 1.4.2 SDK, Oracle 10g Driver, and SQL Server, Sybase or DB2 Database Users” on page 1-27
- “Configurable Directory for Custom Security Provider MBean Types” on page 1-27
- “EJB Features” on page 1-27
- “JDBC Features” on page 1-28
- “User Configuration and User Key Files for `weblogic.Admin` and Ant Tasks” on page 1-30
- “`viewj` and `viewj32` Enhancements” on page 1-31
- “WebLogic Web Services Features” on page 1-31

### Patch Available for Sun Java 2 1.4.2 SDK, Oracle 10g Driver, and SQL Server, Sybase or DB2 Database Users

If you are using WebLogic Server or WebLogic Platform 8.1 SP2 with the Sun Java 2 1.4.2 SDK, or the Oracle 10g driver, or a SQL Server, Sybase or DB2 database, you may need to install the WebLogic Platform 8.1 SP2 SDK1.4.2/Oracle10gdriver/Database patch. This patch, and a description of the specific configurations that require it, are available at the following dev2dev Web site:

[http://dev2dev.bea.com/products/wlplatform81/patch/wlplat81sp2\\_patch.jsp](http://dev2dev.bea.com/products/wlplatform81/patch/wlplat81sp2_patch.jsp)

### Configurable Directory for Custom Security Provider MBean Types

`WL_HOME\server\lib\mbeantypes` is the default directory for installing MBean types. However, if you want WebLogic Server to look for MBean types in additional directories, you can now use the `-Dweblogic.alternateTypesDirectory=<dir>` command-line flag when starting your server.

See the “Install the MBean Type Into the WebLogic Server Environment” sections in *Developing Security Providers for WebLogic Server*.

### EJB Features

WebLogic Server 8.1 Service Pack 2 includes the following new EJB features.

## EJB 2.0 WebLogic QL Extension for EJB QL Provides upper and lower Functions

The WebLogic QL `upper` and `lower` extensions convert the case of arguments to allow finder methods to return results that match the characters in a search expression but not the case. The case change is transient, for the purpose of string matching, and is not persisted in database. The underlying database must also support `upper` and `lower` functions.

### upper

The `upper` function converts characters in its arguments from any case to upper case before string matching is performed. Use the `upper` function with an upper-case expression in a query to return all items that match the expression, regardless of case. For example:

```
select name from products where upper(name)='DETERGENT';
```

### lower

The `lower` function converts characters in its arguments from any case to lower case before string matching is performed. Use the `lower` function with an lower-case expression in a query to return all items that match the expression, regardless of case.

```
select type from products where lower(name)='domestic';
```

## EJBGen Distribution and Examples

As of WebLogic Server 8.1 SP2, EJBGen and associated classes are not included in `weblogic.jar`; instead, they are provided as a separate archive--`ejbgen.jar` in the `WL_HOME/server/lib` of the WebLogic Server distribution. To use EJBGen, put `ejbgen.jar` in your CLASSPATH.

## JDBC Features

WebLogic Server 8.1 Service Pack 2 includes the following new JDBC features.

### BEA WebLogic Type 4 JDBC Drivers for DB2, Informix, Oracle, and Sybase

WebLogic Server 8.1 SP2 includes JDBC drivers from BEA for connecting to DB2, Informix, Oracle, and Sybase databases. The new driver offers JDBC 3.0 compliance and support for some JDBC 2.0 extensions.

See [BEA WebLogic Type 4 JDBC Drivers](#).

## BEA WebLogic Type 4 JDBC Driver for MS SQL Server Enhancements

In WebLogic Server 8.1 SP2, the following enhancements were made to the WebLogic Type 4 JDBC Driver for MS SQL Server:

- Support for retrieval of auto-generated keys
- Improved updatable result set performance
- Improved memory usage for Blob and Clob functionality
- Support for connection failover

## Timeouts for JDBC Connection Creation and Statement Execution

In WebLogic Server 8.1SP2, the following important features were added that help support JDBC connection pools in a high-availability environment:

- `LoginTimeout` connection property supported by the WebLogic Type 4 JDBC drivers which limits the amount of time that a request to create a database connection in a JDBC connection pool will wait for the connection to be created.
- `StatementTimeout` JDBC connection pool attribute which limits the amount of time that a JDBC statement can execute.
- `TestStatementTimeout` JDBC connection pool attribute which limits the amount of time that the JDBC query used to test a database connection can execute. This limit applies to whenever a database connection is tested.

You set the `LoginTimeout` property in the Properties list for a connection pool. After the `LoginTimeout` has elapsed, if the database connection cannot be created, the JDBC driver throws an SQL exception. See “[Limiting Connection Creation Time with LoginTimeout](#)” and connection properties for individual drivers in *WebLogic Type 4 JDBC Drivers*.

You set the `StatementTimeout` and `TestStatementTimeout` attributes in the `config.xml` entry for the JDBC connection pool. The values are in seconds. A value of -1 (the default) indicates that the statements will not timeout. For example:

```
<JDBCConnectionPool Name="testpool"
Password="{3DES}0zvizFP1" Targets="myserver"
InitialCapacity="10" MaxCapacity="10"
DriverName="weblogic.jdbc.oracle.OracleDriver"
Properties="user=john;SID=wls;PortNumber=1433;
serverName=oraclehost;LoginTimeout=60";
```

```
URL="jdbc:bea:oracle://oraclehost:1433"  
StatementTimeout="60" TestStatementTimeout="30"  
</>
```

**Note:** The `StatementTimeout` and `TestStatementTimeout` attributes rely on the JDBC driver implementation of the JDBC specification for `Statement.setQueryTimeout(int seconds)`. Currently, only the WebLogic Type 4 JDBC drivers implement this method.

Also note that there is a known issue regarding this feature with the WebLogic Type 4 JDBC drivers. The drivers currently wait twice as long as specified before timing out the statement. For example, if the `StatementTimeout` is set to 30 seconds, the driver will not time out the statement until 60 seconds has elapsed. This issue is being tracked in CR124744.

## Enhancements to the dbping Java Utility

In WebLogic Server 8.1SP2, the dbping Java utility was enhanced to support the new WebLogic Type 4 JDBC drivers included in this release. See [“dbping”](#) in the WebLogic Server Command Reference for more information.

## Support for oracleConnection Methods

In WebLogic Server 8.1SP2, the following methods were added to support using Oracle Virtual Private Databases:

```
weblogic.jdbc.vendor.oracle.OracleConnection.setClientIdentifier(String)  
weblogic.jdbc.vendor.oracle.OracleConnection.clearClientIdentifier(String)
```

These methods enable you to use Oracle Virtual Private Databases with a connection from a WebLogic JDBC connection pool and without requiring the underlying vendor connection. For more information, see [“Programming with Oracle Virtual Private Databases”](#) and [“Tables of Oracle Extension Interfaces and Supported Methods”](#) in *Programming WebLogic JDBC*.

## Autocommit for XA JDBC Drivers

In WebLogic Server 8.1SP2, by default `autocommit` is set to `false` for all XA JDBC drivers.

## User Configuration and User Key Files for weblogic.Admin and Ant Tasks

For any `weblogic.Admin` command that connects to a WebLogic Server instance, you must provide user credentials. You can now use the new `STOREUSERCONFIG` command to encrypt the user credentials instead of passing credentials directly on the command line or storing unencrypted

credentials in scripts. See “[STOREUSERCONFIG](#)” in the *WebLogic Server Command Reference* for more information.

The `weblogic.Deployer` command, as well as the `wldeploy` and `wlserver` ant tasks, can also use credentials that were encrypted using the `weblogic.Admin STOREUSERCONFIG` command. See the “[Deployment Tools Reference](#)” and “[Using Ant Tasks to Configure a WebLogic Server Domain](#)” for more information.

## viewj and viewj32 Enhancements

Customers expressed interest in using the Tuxedo control to connect a WebLogic application to an existing Tuxedo application that uses VIEWS. To enable this, three new options have been added to the `viewj` and `viewj32` compilers (`weblogic.wtc.jatmi.viewj` and `weblogic.wtc.jatmi.view32`):

```
-bean_names
-compat_names
-modify_strings
```

See “[How to Use the viewj Compiler](#)” in the *WebLogic Tuxedo Connector Programmer's Guide*.

## WebLogic Web Services Features

WebLogic Server 8.1 Service Pack 2 includes the following new Web Services features.

### source2wsdd Meta-Data Tags

When you write the Java class or stateless session EJB that implements a WebLogic Web Service, you can now add optional meta-data tags (identified with the `@wlws` Javadoc tag) that describe what the Web Service looks like. You can then use the `source2wsdd` Ant task to generate the `web-services.xml` file automatically, which will reflect the value of any `@wlws` tags in the Java source file.

See [source2wsdd Tag Reference at http://e-docs.bea.com/wls/docs81/webserv/wlws\\_tags.html](http://e-docs.bea.com/wls/docs81/webserv/wlws_tags.html).

### Examples of Using WebLogic Workshop with WebLogic Web Services

The *Programming WebLogic Web Services* guide has a new section that describes examples and scenarios of using various technologies of WebLogic Platform (the Workshop IDE) to create WebLogic Web Services.

See [Using WebLogic Workshop With WebLogic Web Services at http://e-docs.bea.com/wls/docs81/webserv/wlw.html](http://e-docs.bea.com/wls/docs81/webserv/wlw.html).

## What's New in WebLogic Server 8.1 SP1

This section details major differences between WebLogic Server 8.1 SP1 and earlier versions. This section includes information about the following:

- [“Capacity Planning Documentation” on page 1-32](#)
- [“JDBC Features” on page 1-32](#)
- [“New JSP Tag Added to the Console Extension Tag Library” on page 1-33](#)
- [“Optimized JMS Topic Subscriber Message Selectors” on page 1-34](#)
- [“Specify Non-Anonymous Principals when EJB Container Passivates and Removes Beans” on page 1-34](#)
- [“Web Services Features” on page 1-34](#)

### Capacity Planning Documentation

BEA WebLogic Server runs on hardware ranging from low-end PCs to high-end mainframes. The process of determining what type of hardware and software configuration is required to meet application needs adequately is called capacity planning. This release of WebLogic Server includes the *BEA WebLogic Server Capacity Planning Guide*, which provides capacity planning requirements and uses the 8.1 MedRec application for benchmarking purposes.

### JDBC Features

WebLogic Server 8.1 Service Pack 1 includes the following new JDBC features.

#### **BEA WebLogic Type 4 JDBC Driver for Microsoft SQL Server**

WebLogic Server 8.1 SP1 includes a new JDBC driver from BEA for connecting to a Microsoft SQL Server database. The BEA WebLogic Type 4 JDBC MS SQL Server driver replaces the WebLogic jDriver for Microsoft SQL Server, which is deprecated. The new driver offers JDBC 3.0 compliance, support for some JDBC 2.0 extensions, and better performance. BEA recommends that you use the new BEA WebLogic Type 4 JDBC MS SQL Server driver in place of the WebLogic jDriver for Microsoft SQL Server.

See *BEA WebLogic Type 4 JDBC Drivers*.

## JDBC Connection Pool Property to Handle Erroneous Local Transactions

According to the SQL specification, *any* DDL or DML operation can potentially start a local transaction. It is the application's responsibility to commit or rollback these transactions. Previously, this was not a problem because JDBC drivers did not check for active local transactions before starting a global transaction, or for active global transactions before starting a local transaction. However, the Oracle Thin driver version 920 does check for local and global transactions before starting a new transaction of the other type (global or local), and throws an XAER\_PROTO exception if an application (or WebLogic Server) tries to start a global transaction (XAResource.start()) on a connection when a local transaction is already active on the connection.

To work around this problem, you can set the `rollbackLocalTxUponConnClose` attribute on the JDBC connection pool to `true`. The Weblogic Server will call `rollback()` on the connection to clean up the connection before returning it to the connection pool. Enabling this attribute will have a performance impact because the rollback call requires communication with the database server.

To set the `rollbackLocalTxUponConnClose` attribute, you can use the `setRollbackLocalTxUponConnClose()` method on the `JDBCConnectionPoolMBean` or you can manually add it to the `JDBCConnectionPool` tag in the `config.xml` file. For example:

```
<JDBCConnectionPool
  DriverName="oracle.jdbc.xa.client.OracleXADataSource"
  Name="oraclePool"
  Password="{3DES}8YdvP4FQW3k="
  Properties="user=SCOTT"
  URL="jdbc:oracle:thin:@server:port:sid"
  RollbackLocalTxUponConnClose="true"/>
```

The `RollbackLocalTxUponConnClose` attribute is not available in the Administration Console.

## Support for Oracle Virtual Private Databases

Starting with WebLogic Server 8.1 SP1, WebLogic Server provides support for Oracle Virtual Private Databases (VPDs). A VPD is an aggregation of server-enforced, application-defined fine-grained access control, combined with a secure application context in the Oracle 9i database server.

See [“Programming with Oracle Virtual Private Databases”](#) in *Programming WebLogic JDBC*.

## New JSP Tag Added to the Console Extension Tag Library

A new JSP tag has been added to the Console Extension Tag Library that displays a standard BEA banner in your Administration Console extensions.

See “[<wl:standard-banner> Tag](#)” in *Extending the Administration Console*.

## Optimized JMS Topic Subscriber Message Selectors

For a certain class of applications, WebLogic JMS can optimize topic subscriber message selectors significantly by indexing them. These applications have messages that contain one or more unique identifiers and thousands of subscribers that filter based on these identifiers.

A typical example is an instant messaging application where each subscriber corresponds to a different user, and each message contains a list of one or more target users.

See “[Indexing Topic Subscriber Message Selectors To Optimize Performance](#)” in *Programming WebLogic JMS*.

## Specify Non-Anonymous Principals when EJB Container Passivates and Removes Beans

This release introduces two new security-related elements to the `weblogic-ejb-jar.xml` deployment descriptor. These new elements, `passivate-as-principal-name` and `remove-as-principal-name`, address `ejbPassivate` and `ejbRemove` failures due to security violations.

See “[passivate-as-principal-name](#)” and “[remove-as-principal-name](#)” in *Programming WebLogic Enterprise JavaBeans*.

## Web Services Features

WebLogic Server 8.1 Service Pack 1 includes the following new Web Services features.

### Per Operation Message Security

You can now specify more than one security specification in the `web-services.xml` deployment descriptor file, and associate different security specifications with the operations of a Web Service. Security specifications describe how SOAP messages, generated when a client application invokes a Web Service operation, should be encrypted and digitally signed.

You can also specify that the SOAP request of an invoke of a particular operation use a different security specification than the SOAP response.

See *Associating an Operation With a Particular Security Specification*.

## Timestamps In SOAP Messages

When a client application invokes a WebLogic Web Service configured for message security (encryption or digital signatures), WebLogic Server automatically adds a timestamp to the SOAP response. Additionally, you can configure the Web Service to require a timestamp in the SOAP request, set expiration periods, and so on.

See [Using Timestamps](#).

## Implementing and Assembling a WebLogic Web Service From a Schema File

You can now implement and assemble a WebLogic Web Service starting from an XML Schema file, and preserve the XML Schema throughout the development process so that the WSDL of the deployed Web Service contains the original XML Schema.

See [Assembling a Web Service Starting With an XML Schema](#).

## New Ant Task Features to Support XML Schemas

The `autotype` Ant task now includes the XML Schema in the generated file that contains the data type mapping information. This file is called `types.xml` by default. In previous releases, this generated file contained only the `<type-mapping>` and `<type-mapping-entry>` deployment descriptor elements; now it also includes the `<types>` element.

Similarly, when using the `autotype` or `servicegen` Ant tasks to generate data type components, you can now include a `<types>` element (containing an XML Schema) in the data type mapping file pointed to by the `typeMappingFile` attribute. The Ant tasks merge any generated XML Schemas or data type mapping information with the existing information.

See [autotype](#) or [servicegen](#).

## Mapping Complex Data Types in SOAP Faults to Java Exceptions

WebLogic Server now correctly handles complex data types in service-specific exceptions in accordance with the Java API for XML-Based RPC (JAX-RPC) 1.0 specification.

This is not a new feature per se, but rather a fix in this service pack to the known GA issue CR101160.

## Restricting Access to the WebLogic Web Service Home Page

You can now restrict access to the WebLogic Web Service Home Page by specifying the `exposeHomepage="False"` attribute of the relevant `<web-service>` element in the `web-services.xml` deployment descriptor file.

See *Securing the WSDL and Home Page of the Web Service*.

## **Detailed Overview of Message Security**

The “[Configuring Security](#)” section of the *Programming WebLogic Web Services* guide contains a new section that describes in detail how message security (encryption and digital signatures) works in WebLogic Server.

## What's New in WebLogic Server 8.1

This section details major differences between WebLogic Server 8.1 and earlier versions. Previous sections describe additional changes in Service Pack releases. This section includes information about the following:

- [“Packaging and Deployment Features” on page 1-37](#)
- [“Administration Features” on page 1-40](#)
- [“Sample Applications” on page 1-49](#)
- [“Security Features” on page 1-49](#)
- [“EJB Features and Changes” on page 1-50](#)
- [“J2EE Connector Changes” on page 1-56](#)
- [“JDBC Features and Changes” on page 1-57](#)
- [“JTA Features” on page 1-58](#)
- [“JMS Features” on page 1-59](#)
- [“Web Application Features” on page 1-61](#)
- [“Web Services Features” on page 1-62](#)
- [“Web Service Changed Features In This Release” on page 1-64](#)
- [“WebLogic Tuxedo Connector Features” on page 1-64](#)
- [“XML Features” on page 1-66](#)
- [“Developer Tools” on page 1-66](#)

## Packaging and Deployment Features

WebLogic Server includes the following new and changed features related to J2EE modules and module deployment.

### Split Development Directory Structure

In a development environment, you can use the new WebLogic split development directory structure to build WebLogic Server enterprise applications. Rather than having a single archived EAR file or an exploded enterprise application directory structure, the split development directory structure

provides two parallel directories. The split development directory structure provides several important benefits over the traditional structure, and is accompanied in WebLogic Server by a set of Ant tasks for building, packaging, and deploying applications as traditional EAR files for production use.

See [“Introducing the Split Development Directory Structure”](#) in *Developing WebLogic Server Applications*.

## Custom Classloading for J2EE Modules

You can now create custom classloader hierarchies for an Enterprise Application, allowing for better control over class visibility and reloadability of modules within an .EAR. You achieve this by defining a `classloader-structure` element in the `weblogic-application.xml` deployment descriptor file.

See [“Custom Module Classloader Hierarchies”](#) in *Developing WebLogic Server Applications*.

## New Application Lifecycle Events

You can extend the abstract class `weblogic.application.ApplicationLifecycleListener` to perform application-specific actions when various application lifecycle events occur. WebLogic Server defines the following lifecycle events:

- Initialization—WebLogic Server parses the application’s deployment descriptors before deploying its module(s).
- Preparation—WebLogic Server has identified (and in some cases, started) EJBs, Web Applications, and application-scoped DataSources that are defined in the Enterprise Application.
- Activation—The application is available for processing client requests.
- Update—WebLogic Server detects that one of the application’s deployment descriptors has changed.

You can perform actions for each of the above lifecycle events by supplying the associated method.

See [“Application Lifecycle Events”](#) in *Developing WebLogic Server Applications*.

## Application-Level Class Library

The `APP-INF/lib` directory helps you organize shared class files in an application. WebLogic Server automatically appends classes included in `APP-INF/lib` to the end of the application’s `CLASSPATH`; this ensures that all application modules can access the shared classes.

See [Developing WebLogic Server Applications](#).

## Dynamic Descriptor Changes

WebLogic Server supports updating deployment descriptor attributes in deployed module containers.

## Dynamic Changes to Application-Scoped Pools

WebLogic Server supports updating application-scoped JDBC connection pool properties in the `weblogic-application.xml` deployment descriptor for deployed applications.

## enforceClusterConstraints Option Changed

The `enforceClusterConstraints` option to `weblogic.Deployer` has changed. You now enable or disable cluster constraints for an entire domain either by providing a startup option to the Administration Server or by using an Administration Console control.

See [“Enforcing Cluster Constraints with Two-Phase Deployment”](#) in *Deploying WebLogic Server Applications*.

## Alternate Deployment Descriptors

You can specify an alternate deployment descriptor file to use when deploying an archive file or exploded archive directory. This ability enables you to change the run-time deployment configuration of an application without having to modify and repackage the contents of the archive itself. To use an alternate deployment descriptor, you use one or both of the following options with the `weblogic.Deployer` utility:

- `-altappdd`—specifies the name of an alternate J2EE deployment descriptor, such as `application.xml`.
- `-altwlsappdd`—specifies the name of an alternate WebLogic Server deployment descriptor, such as `weblogic-application.xml`.

See [Deploying WebLogic Server Applications](#).

## No Support for Deploying Unrelated Modules

WebLogic Server no longer supports deploying multiple J2EE modules without an `application.xml` file. If you select a directory for deployment, the directory must contain either a standalone J2EE module (an EJB, Web Application, or Resource Adapter), or multiple modules with an associated `application.xml` file (an Enterprise Application).

## Improved Deployment Performance and Feedback

The speed of J2EE module deployment has improved over previous server versions. You also get additional feedback on the deployment process for both `weblogic.Deployer` and the Administration Console. Feedback is provided by a new JMX notification and filter (`weblogic.management.DeploymentNotification` and `weblogic.management.DeploymentNotificationFilter`), which you can use in your own applications.

See the [Javadocs for WebLogic Classes](#).

## Administration Console Deployment Changes

The WebLogic Server Administration Console interface has been streamlined to make production-level deployment descriptors available for editing by the administrator. Although full deployment descriptor editing is no longer available in the Administration Console, many descriptor elements of interest to Administrators are directly editable via Administration Console fields. You can edit these descriptors without repackaging and redeploying the associated module.

Full J2EE module deployment descriptor editing remains available in the WebLogic Builder application for development use.

The Administration Console also provides new Deployment Assistants to help you deploy different types of J2EE modules. The assistants guide you through the process of selecting deployment files and target servers, and automates the selection of deployment staging modes.

See the [Administration Console Online Help](#).

## Weblogic.Deployer Changes

The `weblogic.Deployer` utility now includes the `-distribute`, `-start`, and `-stop` commands identified in JSR88. Command help has also been reorganized for easy access to basic and advanced commands.

See [Deploying WebLogic Server Applications](#).

## Administration Features

The following new and improved server administration features are included in WebLogic Server 8.1:

## JRockit JVM

This version of WebLogic Server includes BEA WebLogic JRockit 8.1, the first commercial server-side Java Virtual Machine.

See the [JRockit for Windows and Linux User Guide](#) for information about the benefits and usage of JRockit.

## JRockit Monitoring

The Administration Console provides additional runtime data for servers running with the JRockit Virtual Machine (VM).

See [“Monitoring the JRockit Virtual Machine”](#) in the *Administration Console Help*.

## Enhanced Configuration Wizard

The Configuration Wizard is a Java application that creates WebLogic Server administration domain and server configurations. With WebLogic Server 8.1, you can now use the Configuration Wizard to configure such resources as database connectivity (JDBC), messaging services (JMS), and security groups, security roles, and user accounts.

In addition, prior to WebLogic Server 8.1 you could use the Configuration Wizard only to create new domains. Now, you can use the wizard to modify existing domains.

See [Configuring WebLogic Platform](#).

## New Defaults for Development Mode and Production Mode

When you create a domain, you can now indicate whether the domain is to be used in a development environment or a production environment. WebLogic Server uses different default values for various services depending on the type of environment you specify.

See [“Differences Between Configuration Startup Modes”](#) in *Configuring WebLogic Platform*.

## Message Catalogs

Message catalogs are available in HTML format on e-docs as part of the documentation deliverable. You can search for messages by error number using the search engine.

See the [Message Catalog Index](#).

## New Logging APIs

This version of WebLogic Server includes support for the JDK 1.4 logging APIs. With these APIs, you can filter the messages that a server instance writes to its local log file, its standard out, and to the domain-wide message log. Prior to WebLogic Server 8.1, you could filter only the messages that a server instance wrote to the domain-wide log file. You could also establish a minimal level of filtering for a server's output to standard out.

In addition, if you want your applications to receive log messages from WebLogic Server logging services, you can use the JDK 1.4 logging APIs instead of Java Management Extensions (JMX). The JDK 1.4 logging APIs are easier to use than the JMX APIs.

See [“Filtering WebLogic Server Log Messages”](#) and [“Subscribing to Messages”](#) in *Using WebLogic Logging Services*.

## Disabling Remote Exceptions

A new server attribute, `setLogRemoteExceptionsEnabled`, determines whether server message logs include exceptions that are raised in remote systems.

The default value for this attribute is `false`. To change the value of this attribute, use JMX APIs or the following `weblogic.Admin` command:

```
java weblogic.Admin -username username -password password set -mbean
"domain-name:Name=server-name,Type=Server" -property
LogRemoteExceptionsEnabled true
```

For example, on the sample `MedRecServer`:

```
java weblogic.Admin -username weblogic -password weblogic set -mbean
"medrec:Name=MedRecServer,Type=Server" -property
LogRemoteExceptionsEnabled true
```

See the [Javadoc](#) for `weblogic.management.configuration.ServerMBean` and [“weblogic.Admin Command-Line Reference”](#) in *WebLogic Server Command Reference*.

## Timer Service

You can configure the WebLogic Server 8.1 timer service to emit notifications at specific dates and times or at a constant interval. The timer service extends the standard JMX timer service, enabling it to run within a WebLogic Server execute thread and within the security context of a WebLogic Server user account.

See [“Using the WebLogic Timer Service to Generate and Receive Notifications”](#) in *Programming WebLogic Management Services with JMX*.

## Automatic config.xml Archiving

The Administration Server automatically archives old copies of the domain's `config.xml` file when you make changes to the configuration. By default, the Administration Server saves the five most recent versions of `config.xml` in the `/configArchive` subdirectory of the domain. You use the Administration Console to configure the maximum number of archived files to be stored for the domain.

See [“WebLogic Server Archives Previous Versions of config.xml”](#) in *Configuring and Managing WebLogic Server*.

## New Affinity Policies for Java Client Load Balancing

Three new load balancing algorithms minimize the number of IP sockets opened between external Java clients and server instances in a cluster. The new algorithms preserve server affinity by considering a client's existing server connections when accessing objects in a cluster.

The new policies can be applied to EJBs and other RMI objects, as well as to JMS client applications.

See [“Load Balancing in a Cluster”](#) in *Using WebLogic Server Clusters*.

## Improved Node Manager

The Node Manager is a Java application that starts and manages the life cycle of Managed Servers. The following features have been added or improved in WebLogic Server 8.1:

- You can configure the Node Manager process automatically during server installation (using demonstration SSL certificates).
- When you create a Managed Server, WebLogic Server adds your WebLogic Server username and password to the server's Remote Start properties. The Node Manager requires this data to start Managed Servers.
- Improved logging facilities and monitoring capabilities help you better manage multiple servers in a domain.

See [“Configuring, Starting, and Stopping Node Manager”](#) in *Configuring and Managing WebLogic Server*.

## Improved Network Channel Configuration

The functionality of Network Channels has been enhanced to simplify the configuration process. Network Channels now encompass the features that, in WebLogic Server 7.x, required both Network

Channels and Network Access Points. In this version of WebLogic Server, Network Access Points are deprecated.

Network Channels allow you to manage quality of service, meet varying connection requirements, and improve utilization of your systems and network resources. For example, you can use Network Channels to:

- Segregate different types of network traffic
- Support varied application or user requirements on the same Managed Server
- Prioritize network connections that servers use to connect to other servers in a domain

WebLogic Server 8.1 also has new guidelines that apply to configuring Network Channels.

See “[Configuring Network Resources](#)” in *Configuring and Managing WebLogic Server*.

## Changes to Managed Server Independence

Managed Server Independence (MSI) enables Managed Servers to start even if the Administration Server is unavailable.

In previous releases, if a Managed Server could not access an Administration Server, it retrieved its configuration from a file named `config.xml` in the Managed Server's root directory. In WebLogic Server 8.1, a Managed Server retrieves its configuration from a file named `msi-config.xml` in its root directory.

If you enable MSI-replication for a Managed Server, the Administration Server creates the `msi-config.xml` file. This file is a replica of the domain's `config.xml` file.

You can now enable MSI-replication for a Managed Server that shares its root directory with the Administration Server. Before WebLogic Server 8.1, MSI-replication for a Managed Server that shared its root directory with the Administration Server would have overwritten the domain's `config.xml` file with the replica.

## Changes to Discovering Managed Servers

If an Administration Server fails while Managed Servers continue to run, or if you shut down an Administration Server while Managed Servers continue to run, when you restart the Administration Server, it discovers which Managed Servers are running and re-establishes administrative control.

In WebLogic Server 8.1, Administration Servers no longer attempt to discover Managed Servers that have been gracefully or forcefully shut down. If you shut down a Managed Server by killing the JVM or killing the command prompt (shell) in which the server was running, an Administration Server still

attempts to discover the Managed Server and throws an exception when it determines that the Managed Server is no longer running.

If the Administration Server is unable to discover all Managed Servers automatically, you can use a new `weblogic.Admin` command, `DISCOVERMANAGEDSERVER`.

See [“weblogic.Admin Command-Line Reference”](#) in the *WebLogic Server Command Reference*.

## JNDI Names in Deployment Descriptors

WebLogic Server 8.1 deployment descriptors do not support the use of equals signs (=) in JNDI names. For example, the following declaration is no longer supported:

```
<provider-url> ldap://snoopy</provider-url>
<connection-factory-jndi-name>cn=myTest</connection-factory-jndi-name>
```

Prior to this release, the WebLogic Server deployment subsystem would convert equals signs to underscores (\_). Now, the deployment subsystem throws an exception if it encounters a JNDI name with equals signs.

## Administration Console Usability Improvements

The WebLogic Server Administration Console is reorganized to provide better usability for both novice and advanced users. Some of the many Administration Console changes include:

- **Simplified navigation tree** in the left pane of the Administration Console makes it easier to access configuration tabs for server resources and deployed modules.
- **Reorganized server configuration tabs** provide easier access to frequently-used controls.
- **Reorganized SSL and Key Store configuration tabs** facilitate enabling SSL security in a domain.
- **Advanced options toggle** enables you to access less-common, advanced fields by clicking the Advanced button on the relevant page. By default, the Administration Console now displays only the most commonly-used controls on each configuration tab.
- **Improved feedback and exception handling** in the Administration Console provides better feedback on deployment and other administration tasks. It also provides better access to server log files. Within each server log file, you can click on message ID numbers to view more information about an error or exception in the message catalog.
- **New inline help** for Administration Console controls provide help text directly on the configuration tab. You can also access more detailed help for a particular page by using the context-sensitive help icon.

- **Deployment Assistants** in the Administration Console help you deploy different J2EE modules.
- **JDBC Assistants** in the Administration Console help you configure JDBC Connection Pools and DataSources.

For more information about the Administration Console changes, see the [Administration Console Online Help](#).

## weblogic.Admin Features

The `weblogic.Admin` utility provides new commands:

- `BATCHUPDATE` runs multiple `weblogic.Admin` commands in an uninterrupted sequence. (You no longer have to invoke a separate JVM for each `weblogic.Admin` command.)
- `CLUSTERSTATE` returns the number and state of servers in a cluster.
- `DISCOVERMANAGEDSERVER` causes the Administration Server to re-establish administrative control over Managed Servers.
- `QUERY` searches for WebLogic Server MBeans whose `WebLogicObjectName` matches a pattern that you specify.
- `STARTCLUSTER` and `STOPCLUSTER` start and stop all server instances in a cluster.
- `TEST_POOL` tests a connection pool by reserving and releasing a connection from it.
- `VALIDATECLUSTERCONFIG` verifies the formatting of cluster-related attributes in the domain's `config.xml` file.

In addition:

- You can use the new `-adminurl` argument to access runtime MBeans for all server instances through the Administration Server.
- All `weblogic.Admin` commands now return an exit code of 0 if the command succeeds and an exit code of 1 if the command fails.
- The `weblogic.Admin` command now supports the IIOP protocol in addition to T3, T3S, HTTP, and HTTPS.
- The `weblogic.Admin STARTINSTANDBY` command has been removed.

See “[weblogic.Admin Command-Line Reference](#)” in the *WebLogic Server Command Reference*.

## New Default OOTB Performance Parameter Settings

The following performance-related attributes are adjusted to improve the out-of-the-box performance of WebLogic Server. Optimal WebLogic Server production tuning values vary according to your environment and applications.

- The production mode default for the execute queue thread pool size (`ThreadCount`) has been increased from 15 to 25. The development mode default size remains at 15.
- The default maximum capacity of a JDBC connection pool (`MaxCapacity`) now equals the default size of the execute queue thread pool: 25 for production mode; 15 for development mode.
- When native performance packs are used (`NativeIOEnabled=true`), socket reader multiplexor threads now have their own execute queue and do not borrow threads from the default execute queue, which frees up default execute threads to do application work.
- The default execute queue no longer performs internal work (for example, http session invalidation, http session invalidation triggers, and server heartbeat triggers), which frees up execute threads to perform application work.
- The default JDBC connection pool `StatementCacheSize` parameter is now turned on by default and is set to 10. This setting reduces both network roundtrips and preparation work in the database.

## Changes to Default Execute Queue Names

The default execute queue has been renamed to `weblogic.kernel.Default` in this release. The `__weblogic_admin_html_queue` execute queue has been renamed to `weblogic.admin.HTTP`. The `__weblogic_admin_rmi_queue` has been renamed to `weblogic.admin.RMI`.

**Note:** Do not reconfigure or modify the default execute queues.

## New Locations of System Administration Documentation

The system administration documentation has been reorganized for this release. Some of the topics previously included in the *Administration Guide* are now located in the locations indicated in [Table 1-1](#). For a complete index of system administration documentation, see [System Administration](#).

**Table 1-1 New Document Locations**

<b>Topic</b>	<b>New Location</b>
Starting and Stopping Servers	<a href="#">Starting and Stopping Servers</a> in the <i>Administration Console Online Help</i>
Using Log Messages to Manage WebLogic Server	<a href="#">Server Log</a> in the <i>Administration Console Online Help</i>
Deploying Applications	<a href="#">Deploying Applications and Modules</a> in the <i>Administration Console Online Help</i>
Managing Transactions	<a href="#">JTA</a> in the <i>Administration Console Online Help</i>
JDBC	<a href="#">JDBC, JDBC Connection Pools, JDBC DataSources, JDBC MultiPools</a> in the <i>Administration Console Online Help</i>
JMS	<a href="#">Configuring JMS, Tuning JMS, and Monitoring JMS</a> in the <i>Administration Console Online Help</i>
Messaging Bridge	<a href="#">Messaging Bridge</a> in the <i>Administration Console Online Help</i>
JNDI	<a href="#">JNDI</a> in the <i>Administration Console Online Help</i>
J2EE Connectors	<a href="#">Connectors</a> in the <i>Administration Console Online Help</i>
Managing WebLogic Server Licenses	<a href="#">Installing and Updating WebLogic Platform License Files</a> in <i>Installing WebLogic Platform</i> .
Java Utilities	<a href="#">Using the WebLogic Server Java Utilities</a> in the <i>WebLogic Server Command Reference</i>
Command-Line Interface (weblogic.Admin)	<a href="#">weblogic.Admin Command-Line Reference</a> in the <i>WebLogic Server Command Reference</i> .
Configuring Network Resources	<a href="#">Configuring and Managing WebLogic Server</a>
Overview of WebLogic Server Domains	<a href="#">Configuring and Managing WebLogic Server</a>
Creating New Domains Using the Configuration Wizard	<a href="#">Configuring and Managing WebLogic Server</a>

**Table 1-1 New Document Locations (Continued)**

Topic	New Location
Recovering Failed Servers	<a href="#">Configuring and Managing WebLogic Server</a>
Node Manager	<a href="#">Configuring and Managing WebLogic Server</a>
Monitoring a WebLogic Server Domain	<a href="#">Configuring and Managing WebLogic Server</a>

## Sample Applications

A new J2EE sample application, Avitek Medical Records (or MedRec), concisely demonstrates all aspects of the J2EE platform. Designed as an educational tool for all levels of J2EE developers, MedRec showcases the use of each J2EE component, and illustrates best practice design patterns for component interaction and client development. Medical Records is available from the Start menu on Windows machines. On Linux and other platforms it can be started from the `WL_HOME\samples\server\config\medrec` directory.

WebLogic Server sample applications have also been updated to use the PointBase® version 4.3 server as the sample datastore.

## Security Features

The following new and improved security features are included in WebLogic Server 8.1.

### Improved Functionality for Creating Roles and Policies

New windows and improved options facilitate managing access to WebLogic resources such as the Administration Console, the `weblogic.Admin` tool, MBeans, applications, COM, EIS, EJB, JDBC, JNDI, JMS, servers, and Web applications.

See “[Security](#)” in the *Administration Console Online Help*.

### Improved Support for Keystores and SSL Configuration

The SSL implementation of WebLogic Server supports the use of keystores for storing private keys and trusted CAs. Keystores add a level of protection to the flat files used in past release of WebLogic Server.

The default configuration of SSL and demonstration keystores provide users with secure communication out of the box. The configuration of keystores and SSL for a production environment has been simplified by the implementation of an assistant.

See [“Configuring Keystores and SSL”](#) in the *Administration Console Online Help*.

## Support for the Sun Java Cryptography Extension (JCE) Package

The Java Cryptography Extension (JCE) is a set of packages that provide a framework for encryption using strong ciphers, key generation and agreement, and Message Authentication Code algorithms.

See [“WebLogic Server Security Service”](#) in *Introduction to WebLogic Server and WebLogic Express*.

## EJB Features and Changes

This release of WebLogic Server introduces the following EBJ features and changes.

### appc Compiler

`appc` is a single tool for compiling and validating a J2EE ear file, an ejb-jar file or war file for deployment. Previously, a user wanting to compile all modules within an ear file had to extract the individual components of an ear and manually execute the appropriate compiler (`jspc` or `ejbc`) to prepare the module for deployment. `appc` automates this process and performs additional pre-deployment validation checks not previously performed.

`appc` is discussed in detail in [“appc”](#) in *Programming WebLogic Enterprise JavaBeans*.

### Batch Operations

WebLogic Server now supports batch updates and deletes, in addition to the existing batch insert (previously known as “bulk insert”) support. In addition, the EJB container now prevents exceptions by performing dependency checks between batch operations.

See [“Batch Operations”](#) in *Programming WebLogic Enterprise JavaBeans*.

### Automatic Table Creation Feature Expanded

Previously, WebLogic Server automatically created database tables if the `create-default-dbms-tables` element in `weblogic-cmp-rdbms-jar.xml` was set to `True`. However, if the table already existed, the server did not recreate it. Beginning with this release, WebLogic Server recreates existing tables when underlying table schema change, that is, when any container-managed persistence fields are modified.

See [“Automatic Table Creation”](#) in *Programming WebLogic Enterprise JavaBeans*.

## Preventing Deadlocks for Container-Managed Persistence Beans

In situations of high throughput, applications that use an exclusive concurrency strategy can encounter deadlocks if a transaction that performs a cascade delete needs access to the same bean as a transaction that does not perform a cascade delete. This release includes a feature to prevent deadlocks in this scenario, via the `lock-order` element in the `weblogic-cmp-rdbms-jar.xml` deployment descriptor file.

See [“Preventing Deadlocks for Transactions that Use Exclusive Concurrency”](#) in *Programming WebLogic Enterprise JavaBeans*.

## Fine-Grained EJB Redeployment

During iterative development of an EJB application, developers make many modifications to EJB implementation class files; they typically redeploy an EJB module multiple times during its development.

Previously, in order to redeploy an implementation class, developers had to reload the entire EJB module containing the implementation class, including all other implementation classes as well as utility classes. With this new feature, developers can specify reloading granularity to the level of an individual implementation class.

See [“WebLogic Application Classloading”](#) in *Developing WebLogic Server Applications* for a detailed explanation of this feature.

## EJB Query Language (QL) Compiler Enhancements

Compiler error messages in EJB QL now provide a visual aid to identify which part of the query is in error and allow the reporting of more than one error per compilation.

See [“EJB QL Error-Reporting Enhancements”](#) in *Programming WebLogic Enterprise JavaBeans*.

## Performance Improvements

WebLogic Server provides improved performance for EJB bulk updates, optimistic concurrency, field groups, relationship caching, and EJB redeployment.

This release also introduces greatly improved monitoring of performance, via new tab pages in the WebLogic Server Administration Console.

See [“Tuning WebLogic Server EJBs”](#) in *WebLogic Server Performance and Tuning*.

## Reloadable EJB Modules

With the Reloadable J2EE Modules feature, you can also redeploy EJB modules independently of other components in an Enterprise Application.

See [Developing WebLogic Server Applications](#).

## EJB Deployment Assistants

The Administration Console provides an EJB Module Deployment Assistant to help you deploy EJBs. See the [EJB section of the Administration Console Online Help](#).

## New Default Values for Several EJB-Specific Deployment Descriptor Elements

Several EJB-related deployment descriptor elements now have new default values, in compliance with the J2EE 1.3 specification.

**Table 1-2 New Default Values for EJB Deployment Descriptor Elements**

Deployment Descriptor File	Element	New Default Value
<code>weblogic-ejb-jar.xml</code>	<code>enable-call-by-reference</code>	False
<code>weblogic-cmp-rdbms-jar.xml</code>	<code>check-exists-on-method</code>	True
	<code>include-updates</code>	True

For a complete discussion of `weblogic-ejb-jar.xml` elements, see [“The weblogic-ejb-jar.xml Deployment Descriptor”](#) in *Programming WebLogic Enterprise JavaBeans*.

For a complete discussion of `weblogic-cmp-rdbms-jar.xml` elements, see [“The weblogic-cmp-rdbms-jar.xml Deployment Descriptor”](#) in *Programming WebLogic Enterprise JavaBeans*.

## New dbms-column-type Values

WebLogic Server supports two additional values for the `dbms-column-type` element in `weblogic-cmp-rdbms.xml`: `LongString` and `SybaseBinary`.

See [“dbms-column-type”](#) in *Programming WebLogic Enterprise JavaBeans*.

## Relationship Caching Functionality Allows Multiple Caching Elements

The  `caching-element`  tag in  `weblogic-cmp-rdbms-jar.xml`  is used in relationship-caching to specify the  `cmr-field`  for the related bean, and the  `group-name`  in the related bean.

The WebLogic Server EJB container now allows multiple  `caching-element`  sub-elements. The relevant DTD entry is:

```
<!ELEMENT caching-element (
  cmr-field,
  group-name?,
  caching-element*
)>
```

Previously, the DTD entry read this way:

```
<!ELEMENT caching-element (
  cmr-field,
  group-name?,
  caching-element?
)>
```

## Disabling Deployment Warning Messages

A new feature can disable certain warning messages during deployment, via the new  `disable-warning`  element in  `weblogic-ejb-jar.xml` .

See [“Disabling Deployment Warning Messages”](#) in *Programming WebLogic Enterprise JavaBeans*.

## New Security-related Deployment Descriptor Elements

Two new security-related deployment descriptor elements have been added to  `weblogic-ejb-jar.xml` :

- The  `externally-defined`  element replaces  `global-role` , which has been deprecated in this release. See [“externally-defined”](#) in *Programming WebLogic Enterprise JavaBeans*.
- The  `run-as-role-assignment`  element is used to map a given  `security-identity/ /run-as/role-name`  that is specified in the  `ejb-jar`  deployment descriptor to a  `run-as-principal-name` . See [“run-as-role-assignment”](#) in *Programming WebLogic Enterprise JavaBeans*.

For a comprehensive discussion of these elements and their relationship to securing of EJBs in general, see [“Securing Enterprise JavaBeans”](#) in *Programming WebLogic Security*.

## Constraining Optimistic Row Checking

To specify the rows in a table that the EJB container should check when optimistic concurrency is used, use the new `verify-rows` element in `weblogic-cmp-rdbms-jar.xml`. This element allows you to choose between performing optimistic checking on any row read by a transaction versus only rows updated or deleted by a transaction.

See [“verify-rows”](#) in *Programming WebLogic Enterprise JavaBeans*.

## Storing Table Creation Scripts

To specify the DDL file name to which the EJB container writes table creation scripts, use the new `default-dbms-tables-ddl` element in `weblogic-cmp-rdbms-jar.xml`.

See [“default-dbms-tables-ddl”](#) in *Programming WebLogic Enterprise JavaBeans*.

## Assigning a Message-Driven Bean to a Configured Execute Queue

As of this release, you can assign message-driven beans to a configured execute queue, via the `dispatch-policy` element in `weblogic-ejb-jar.xml`. Previously, `dispatch-policy` applied only to session and entity beans.

While you can also set the dispatch policy via appc's `-dispatchPolicy` flag, BEA strongly recommends you use the deployment descriptor element instead. This way, if the EJB is recompiled—during deployment for example—the setting will not be lost.

See [“The weblogic-ejb-jar.xml Deployment Descriptor”](#) in *Programming WebLogic Enterprise JavaBeans*.

## EJB Features Deprecated in This Release

The following EJB features have been deprecated in this release.

### Administration Console Deployment Descriptor Editor

This release of WebLogic Server deprecates the Administration Console Deployment Descriptor Editor.

The majority of deployment descriptor elements can now only be edited using a text or XML editor. A small subset of elements—those related to administering and tuning EJBs—can still be viewed, modified, and persisted to the descriptor file via the Administration Console using the Descriptors tab.

See “[Configuring Deployment Descriptor Values](#)” in the EJB section of the *Administration Console Online Help*.

### ejbc Compiler

The `ejbc` compiler has been deprecated. Use `appc` in its place. See “[appc](#)” in *Programming WebLogic Enterprise JavaBeans*.

### sql-select-distinct

This version of WebLogic Server deprecates the `sql-select-distinct` element in `weloblogic-cmp-rdbms-jar.xml`. Use the `DISTINCT` clause directly in finder queries instead of this deployment descriptor element. For finder queries that have a `DISTINCT` clause, the container defers duplicate elimination to the database if `FOR UPDATE` is not used and filters duplicates in memory if it is used.

### global-role

This version of WebLogic Server deprecates the `global-role` element in `weloblogic-ejb-jar.xml`. Use the `externally-defined` element instead. For a comprehensive discussion of the `externally-defined` element and its relationship to securing of EJBs in general, see “[Securing Enterprise JavaBeans](#)” in *Programming WebLogic Security*.

### run-as-identity-principal

This version of WebLogic Server deprecates the `run-as-identity-principal` element in `weloblogic-ejb-jar.xml`. Use the `run-as-principal-name` element instead. For a comprehensive discussion of the `run-as-principal-name` element and its relationship to

securing of EJBs in general, see [“Securing Enterprise JavaBeans”](#) in *Programming WebLogic Security*.

## stateless-bean-methods-are-idempotent

This version of WebLogic Server deprecates the `stateless-bean-methods-are-idempotent` element in `weblogic-ejb-jar.xml`. Use the `idempotent-methods` element instead. See [“idempotent-methods”](#) in *Programming WebLogic Enterprise JavaBeans*.

## J2EE Connector Changes

The DTD for the WebLogic Server Connector deployment descriptor, `weblogic-ra.xml`, has changed in this version. See [“weblogic-ra.xml Deployment Descriptor Elements”](#) in *Programming WebLogic Server J2EE Connectors*.

The Connector implementation now makes connections shareable unless a suitable “hint” is specified at deployment time. EJB 2.0 uses the `res-sharing-scope` deployment descriptor (with values `Shareable` or `Unshareable`) to specify this hint.

## jCOM Features

In this release, you configure COM packet timeout values and the maximum length of COM message packets via a different location in the Administration Console.

These changes are summarized in the following table.

**Table 1-3 Changes to COM Packet Configuration Values**

Value	Configured This Way in 7.0	Configure This Way in 8.1
COM packet timeout value	Set the <code>COM Message Timeout</code> property under <code>Server -&gt; Connections -&gt; jCOM</code> in the Administration Console.	Set the <code>Complete Message Timeout</code> property under <code>Server -&gt; Protocols -&gt; General -&gt; Advanced Options</code> in the Administration Console.
The maximum length of COM message packets	Set the <code>COM Max Message Size</code> property under <code>Server -&gt; Connections -&gt; jCOM</code> in the Administration Console.	Set the <code>Complete Maximum Message Size</code> property under <code>Server -&gt; Protocols -&gt; General -&gt; Advanced Options</code> in the Administration Console.

## JDBC Features and Changes

WebLogic Server provides the following new JDBC features, along with internal performance enhancements in the JDBC subsystem.

### JDBC Assistants

The Administration Console includes the JDBC Connection Pool Assistant and the JDBC Data Source Assistant. These assistants help ease database connectivity configuration by prompting you for database, JDBC driver, and connection pool information, and then constructing the connection attributes required by your JDBC driver. See the [Administration Console Online Help](#).

### JDBC Connection Pool Attributes

JDBC connection pools include several new attributes and features that you can configure from the Administration Console or MBean attributes using the JMX API, including among others:

- **Connection Reserve Timeout**—Enables connection requests to wait for a connection from a connection pool when all connections are currently in use.
- **Connection Creation Retry Frequency**—Enables WebLogic Server to retry to create a database connection after the original attempt to create the connection failed.
- **Test Created Connections**—Enables testing and initialization of physical database connections when each connection is created.
- **Test Pool**—Tests a JDBC connection pool by reserving and releasing a connection.

See the [Administration Console Online Help](#).

### Enhanced Support for JDBC Extensions

Some database vendors provide additional proprietary methods for working with data from their DBMS. These methods extend the standard JDBC interfaces. WebLogic Server provides enhanced support for vendor extensions to JDBC by supporting most extension methods exposed in a public interface in the vendor's JDBC driver. See [Using Vendor Extensions to JDBC Interfaces](#) in *Programming WebLogic JDBC*.

### Physical Connection from a Connection Pool

When you get a connection from a connection pool, WebLogic Server provides a logical connection rather than a physical connection so that WebLogic Server can manage and maintain the connection. In some cases, you may want to use a physical connection, such as if you need to pass the connection to a method that checks the class name of the object for a particular class. WebLogic Server includes

the `getVendorConnection()` method in the `weblogic.jdbc.extensions.WLConnection` interface that you can use to get the underlying physical connection from a logical connection. See [Getting a Physical Connection from a Connection Pool](#) in *Programming WebLogic JDBC*.

## Support for RowSets

WebLogic Server includes support for RowSets, which are a JDBC 2.0 extension to ResultSets. RowSets allow a user to read and modify a cached query result and then commit the resulting changes back to a database. RowSets use a disconnected model which uses optimistic concurrency control to ensure database consistency. This allows work to be performed without holding open long transactions or database and application server resources. See [Using RowSets with WebLogic Server](#) in *Programming WebLogic JDBC*.

## Statement Caching

The statement cache for JDBC connection pools was enhanced to include a Least Recently Used caching algorithm and controls for clearing the statement cache.

When you use a prepared statement or callable statement in an application or EJB, there is considerable processing overhead. To minimize the processing costs, WebLogic Server can cache statements used in your applications in the statement cache. When an application or EJB calls any of the statements stored in the cache, WebLogic Server reuses the statement stored in the cache, which reduces CPU usage on the database server, and thus improves performance for the current statement and leaves CPU cycles on the database server for other tasks. See the [Administration Console Online Help](#).

## Classes Removed from `weblogic.jdbc.pool`

In WebLogic Server 8.1, the `weblogic.jdbc.pool` classes were removed, except for the `weblogic.jdbc.pool.Driver` class. These classes were removed because they were incompatible with an internal change that enables and enhances support for JDBC extensions provided in JDBC drivers. For more information, see [“Removed Classes”](#) in the *Upgrade Guide*.

## JTA Features

WebLogic Server 8.1 provides the following new JTA features.

## Manual Completion of Current Transactions

In some cases, a transaction may not complete normally due to system or network failures. In such situations there may be locks held on behalf of the pending transaction that are inhibiting the

progress of other transactions. You can use the Administration Console or methods on the JTA runtime MBean to manually complete transactions that did not complete normally. See the [Administration Console Online Help](#).

## Non-XA-Compliant Resource Participation in a Global Transaction

A single, non-XA-compliant resource adapter can participate in a global transaction with other XA-compliant resources. WebLogic Server uses a last agent commit optimization so that after all participating XA-compliant resources are prepared, the result of the local transaction for the non-XA resource is used to determine the outcome of the global transaction. The resource adapter must provide local transaction semantics. You can use this functionality with the WebLogic Server J2EE Connector architecture to enable non-XA legacy systems to participate in a global transaction.

## JMS Features

WebLogic Server 8.1 provides the following new JMS features.

### JMS Thin Client

At approximately 400KB, the JMS thin application client (`wljmsclient.jar`) file provides full WebLogic JMS functionality, yet greatly reduces the client-side WebLogic footprint by using a smaller library that contains only the set of supporting files required by client-side programs. The JMS thin client also requires using the standard WebLogic thin application client JAR (`wlclient.jar`), around 300KB, which contains the base client support for clustering, security, and transactions, and failover. See [WebLogic JMS Thin Client](#) in *Programming WebLogic JMS*.

### Simplified Access to Remote WebLogic Domains or Third-Party JMS Providers

Using the Foreign JMS Server node on the Administration Console, you can quickly map a third-party JMS provider so that its connection factories and destinations appear in your WebLogic Server JNDI tree as a local JMS objects. A Foreign JMS Server configuration can also be used to reference remote instances of WebLogic Server in another cluster or domain in your local WebLogic JNDI tree. See [Accessing Foreign JMS Providers](#) in the *Administration Console Online Help*.

### Easier Access to JMS Via EJBs and Servlets

New “wrappers” for JMS connection factories make it easier to use JMS inside a J2EE container, such as an EJB or servlet. By using a `resource-ref` element in the deployment descriptors to define your connection factories, the wrappers make it easy to access local or remote WebLogic JMS objects, and

can even be used to access third-party JMS providers. This feature also provides automatic pooling of JMS connection and session objects (and some pooling of message producer objects as well); automatic transaction enlistment for JMS providers that support XA; monitoring of the JMS connection and re-establishment after a failure; and security credentials that are managed by the EJB or servlet. See [Using JMS with EJBs and Servlets](#) in *Programming WebLogic JMS*.

## Better Expired Message Handling

Active message expiration ensures that expired messages are cleaned up immediately. Moreover, expired message auditing gives you the option of tracking expired messages, either by logging when a message expires or by redirecting expired messages to a special destination. See [Handling Expired Messages](#) in the *Administration Console Online Help*.

## Improved Message Flow Control by Blocking Producers

The “Blocking Send” features help you avoid receiving message quota errors by temporarily blocking message producers from sending messages to a destination (queue or topic) when the destination has exceeded its specified maximum message quota. See [Avoiding Quota Exceptions by Blocking Message Producers](#) in the *Administration Console Help*.

## Ordered Redelivery of Messages

As per the JMS Specification, all messages initially delivered to a consumer from a given producer are guaranteed to arrive at the consumer in the order in which they were produced. WebLogic JMS goes above and beyond this requirement by guaranteeing the correct ordering of redelivered messages as well. See [Ordered Redelivery of Messages](#) in *Programming WebLogic JMS*.

## Dynamically Deleting Queue or Topic Destinations

New JMS Helper extension methods enable you to delete JMS destinations dynamically. The JMS server removes the deleted destination in real time; therefore, it is not necessary to redeploy the JMS server for the deletion to take effect. See [Deleting Destinations Dynamically](#) in *Programming WebLogic JMS*.

## ServerSessionPoolFactory Class Deprecated In `weblogic.jms` Package

The `ServerSessionPoolFactory` class in the `weblogic.jms` package has been deprecated in WebLogic Server 8.1. It has been replaced by the `ServerSessionPoolFactory` class in the `weblogic.jms.extensions` package. BEA recommends using the new version in the `weblogic.jms.extensions` package when binding a `ServerSessionPoolFactory` into JNDI. However, for this release you can still perform the JNDI lookup with either version. See [Defining](#)

[Server Session Pools](#) in *Programming WebLogic JMS* or the `weblogic.jms.extensions.ServerSessionPoolFactory` Javadoc.

## Web Application Features

WebLogic Server 8.1 includes the following new Web Application features and changes.

### Performance Improvements

Performance is improved for JSP string handling as well as JSP compilation time.

### Security Principle for init Method

Use the new `init-as-principal-name` element in `weblogic.xml` to declare a principle name for running a servlet's `init` method. See [Developing Web Applications for WebLogic Server](#).

### Allow Access to WEB-INF on forward/include

You can now call one servlet from inside another servlet. This is accomplished using either a forward or an include request from within the original servlet. Should you forward to a second servlet, all future action takes place according to the second servlet, as with any forward. Including a second servlet allows you to gather data from a source already accessed by another servlet without having to rewrite all the code. See [Dispatching Requests to Another Resource](#) in *Programming WebLogic HTTP Servlets*.

### Integrated jspc Functionality into appc

The `appc` compiler now incorporates the functionality of `jspc`. You can use `appc` to compile and generate both EJBs and JSPs for deployment. See [appc and jspc Compilers](#) in *Developing Web Applications for WebLogic Server*.

### Class Reloading for Servlet Filters

When responding to a request for a servlet, WebLogic Server checks the time stamp of the servlet class file prior to applying any filters associated with the servlet, and compares it to the servlet instance in memory. If a newer version of the servlet class is found, WebLogic Server re-loads the servlet class before any filtering takes place. You can configure the interval at which WebLogic Server checks the timestamp using the `Server Reload` attribute. See [Servlet Development Tips](#) in *Programming WebLogic HTTP Servlets*.

## FileServlet File Sorting Options

WebLogic Server introduces new `weblogic.xml` deployment descriptors to provide sorting options for directory listings. The new element, `index-directory-sort-by` has valid sorting styles of `NAME`, `LAST_MODIFIED`, and `SIZE`. For example, to enable directory listing sorted by file size, the XML would look similar to:

```
<weblogic-web-app>
  <index-directory-enabled>true</index-directory-enabled>
  <index-directory-sort-by>SIZE</index-directory-sort-by>
</weblogic-web-app>
```

## Web Services Features

The following new Web Services features are available with WebLogic Server 8.1.

### Digital Signatures and Encryption

You can configure data security for Web Services and Web Service clients using new elements in the `web-services.xml` deployment descriptor. See [Configuring Security](#) in *Programming WebLogic Web Services*.

### Reliable SOAP Messaging

Reliable SOAP messaging is a framework whereby an application running in one WebLogic Server instance can asynchronously and reliably invoke a Web service running on another WebLogic Server instance. See [Using Reliable SOAP Messaging](#) in *Programming WebLogic Web Services*.

### SOAP 1.2

WebLogic Server supports SOAP 1.2 as the message transport when a client invokes a Web Service operation. See [Using SOAP 1.2](#) in *Programming WebLogic Web Services*.

### JMS Transport Protocol

You can optionally configure a Web Service to use JMS as the transport protocol (in addition to HTTP/S, the default protocol) when a client accesses the service. See [Using JMS Transport to Invoke a WebLogic Web Service](#) in *Programming WebLogic Web Services*.

## Asynchronous Invocation of WebLogic Web Services

The `clientgen` Ant task can now generate stubs for invoking a Web service operation asynchronously. The stub contains two methods: the first invokes the operation with the required parameters but does not wait for the result; later, the second method returns the actual results. You use this asynchronous client when using reliable SOAP messaging. See [Writing an Asynchronous Client](#) in *Programming WebLogic Web Services*.

## Portable Stubs

You can now use portable stubs (versioned client JAR files used to invoke WebLogic Web services) to avoid class clashes when invoking a Web service from within WebLogic Server. See [Creating and Using Portable Stubs](#) in *Programming WebLogic Web Services*.

## Implementation of the SOAP With Attachments API For Java (SAAJ) 1.1

SAAJ enables developers to produce and consume messages conforming to the SOAP 1.1 specification and SOAP with Attachments note. This specification is derived from the `java.xml.soap` package originally defined in the JAXM 1.0 specification.

## wsdlgen Ant Task

The `wsdlgen` Ant task generates a WSDL file from the EAR and WAR files that implement your Web Service. The EAR file contains the EJBs that implement your Web Service and the WAR file contains the `web-services.xml` deployment descriptor file. See [Web Service Ant Tasks and Command-Line Utilities](#) in *Programming WebLogic Web Services*.

## Additional Functionality in Existing Ant Tasks

The existing Web Service Ant tasks have the following new attributes and elements:

- `autotype`—The `autotype` Ant task has the following new attributes: `keepGenerated` and `overwrite`.

The Ant task also has a new command-line version.

- `clientgen`—The `clientgen` Ant task has the following new attributes: `generateAsyncMethods`, `generatePublicFields`, and `keepGenerated`.
- `servicegen`—The `servicegen` Ant task has the following new child elements: `<handlerChain>`, `<reliability>`, and `<security>`.

The main `servicegen` Ant task has the following new attributes: `keepGenerated` and `mergeWithExistingWS`.

The `<service>` child element of the `servicegen` Ant task has the following new attribute: `useSoap12`.

- The `source2wsdd` Ant task has the following new attributes: `ejblink`, `mergeWithExistingWS`, `overwrite`, and `wsdlFile`.
- The `wsdl2Service` Ant task has the following new attribute: `overwrite`.
- The `wspackage` Ant task has the following new attribute: `utilJars`.

For detailed information, see [Web Service Ant Tasks and Command-Line Utilities](#) in *Programming WebLogic Web Services*.

## Web Service Changed Features In This Release

The following Web Service features have changed in this release.

### **wsdl2Service Ant Task**

The `wsdl2Service` Ant task now generates a Java interface that represents the implementation of your Web Service. In the previous release, this Ant task generated a Java source file that partially implemented the Web service.

### **JMS-Implemented Web Services**

BEA no longer supports using JMS Topics as the destination type for JMS-implemented WebLogic Web Services. This functionality was deprecated in release 7.0 of WebLogic Server.

### **Command-Line Version of the `servicegen` Ant Task**

BEA no longer supports the command-line version of the `servicegen` Ant task.

## WebLogic Tuxedo Connector Features

WebLogic Server 8.1 provides the following new WebLogic Tuxedo Connector features.

### **Enhanced Security Administration**

WebLogic Tuxedo Connector provides to users the ability to select one of the following APPKEY generators to access Tuxedo services:

- TPUSER plug-in—Enables users to use a `tpuser` file to provide user information to the Tuxedo authentication server.
- LDAP—Enables users create a single source of security administration by allowing WebLogic Server embedded LDAP information to be used by a Tuxedo 8.1 authentication server.
- Custom—Enables users to create a custom APPKEY to generate user information to access Tuxedo services.

## WLEC to WebLogic Tuxedo Connector Migration Guide

The guide outlines how to migrate applications using WLEC to the WebLogic Tuxedo Connector. WebLogic Tuxedo Connector provides support for FactoryFinder objects using the `find_one_factory_by_id` method. WLEC to WebLogic Tuxedo Connector migration requires minor application modification:

- WLEC applications require modification of the portions of application code that use or call environmental objects.
- Existing CORBA C++ server objects do not require server application changes.

See the [WLEC to WebLogic Tuxedo Connector Migration Guide](http://e-docs.bea.com/wls/docs81/wlec_migration/index.html) at [http://e-docs.bea.com/wls/docs81/wlec\\_migration/index.html](http://e-docs.bea.com/wls/docs81/wlec_migration/index.html).

## Asynchronous tpacall

The asynchronous `tpacall` method allows you to send a request to a Tuxedo service and release the thread resource that performed the call to the thread pool. This allows a very large number of outstanding requests to be serviced with a much smaller number of threads. See [Request/Response Communication](#) in the *WebLogic Tuxedo Connector Programmer's Guide*.

## Runtime WTC ORB

This release of WebLogic Tuxedo Connector implements a new WTC ORB which uses WebLogic Server RMI-IIOP runtime and CORBA support. Previous releases used a JDK based WTC ORB. A wrapper is provided to allow users with legacy applications to use the new WTC ORB without modifying their existing applications. See [How to Develop WebLogic Tuxedo Connector Client Beans using the CORBA Java API](#) in the *WebLogic Tuxedo Connector Programmer's Guide*.

## Enhancements to FML Interface

Several performance enhancements are provided in the Java Application-to-Transaction Monitor Interface Field Markup Language ([JATMI FML Interface](#)). Enhancements include:

- The addition of a `Fadd` method to add objects to a fielded buffer
- Performance enhancements for the `Fchg` and `Fdel` methods

## VIEW Buffer Example

The `simpview` example demonstrates the ability of WebLogic Tuxedo Connector to allow WebLogic Server to interoperate with Tuxedo using VIEWS. WebLogic Tuxedo Connector code examples, if installed, are located in the `SAMPLES_HOME\server\examples\src\examples\wtc` directory of your WebLogic Server installation, where `SAMPLES_HOME` is the location of all examples for the WebLogic Platform.

## Changes to WTC in the Navigation Tree

The terminology used to reference the WebLogic Tuxedo Connector Service and nodes in Administration Console has changed:

- The WebLogic Tuxedo Connector node is now WTC.
- The Local WLS Domains node is now Local Tuxedo Access Points.
- The Remote Tuxedo Domains node is now Remote Tuxedo Access Points.
- A WTCServer instance is now a WTC Service instance.

## XML Features

You can now use the WebLogic XPath API to perform XPath matching against an XML document represented as a DOM, XMLNode, or XMLInputStream. See [Using the WebLogic XPath API](#) in *Programming WebLogic XML*.

## Developer Tools

The following new developer tool features are available in WebLogic Server 8.1.

### New appc Compiler

The appc compiler compiles and generates EJBs and JSPs for deployment. It also validates the descriptors for compliance with the current specifications at both the individual module level and the application level. The application level checks include checks between the application-level deployment descriptors and the individual modules as well as validation checks across the modules.

## New J2EE Client .JARs

Prior to version 8.1, client applications that incorporated WebLogic Server functionality required the entire WebLogic Server distribution (`weblogic.jar` and `weblogicaux.jar`) on the client machine. WebLogic Server now provides two new client `.jar` files that include only the functionality needed for small-footprint J2EE client functionality. The new files are:

- `wlclient.jar` for basic WebLogic functionality such as clustering, security, and transactions.
- `wljmsclient.jar` for basic WebLogic functionality plus JMS features.

The new client `.jar` files are located in the `/server/lib` subdirectory of the WebLogic Server installation directory (for example, `c:\bea\weblogic81b\server\lib`). The new client `.jar` files are not supported with JDK 1.3.x or earlier.

See [Developing WebLogic Server Applications](#).

## Builder Changes

The following features and changes apply to the WebLogic Builder tool.

### Deployment Descriptor Editing

Use the Builder tool to edit J2EE module deployment descriptors. Deployment Descriptor editing features are no longer available via the Administration Console. See the [WebLogic Builder Online Help](#).

### Optimistic Concurrency

It is now possible to configure your CMP entity beans to use optimistic concurrency for parallel transactions using Builder. See [Working with EJBs](#) in the [WebLogic Builder Online Help](#).

## Internationalization Features

WebLogic Server introduces the following changes to internationalization utilities:

- `weblogic.i18ngen` has updated command line options.
- `weblogic.i10ngen` has updated command line options.
- `weblogic.gettxt` is a new command line utility.
- `weblogic.i18ntools.GetText` is a new API.

- `weblogic.MsgEditor` has an updated GUI. The main Message Editor window also provides the ability to retire and unretire messages. Retiring a message does not mean that the message is deleted from the master catalog. It simply means it is hidden from user view. This feature is useful for removing obsolete messages. If you need to bring a retired message back into view, you can unretire it.

## Certifications

For the latest information about supported WebLogic Server configurations, see [Supported Configurations](#).

## Standards Support

The following tables provide information about J2EE and other standards supported in WebLogic Server 8.1:

- Java Standards Support: [Table 1-4](#)
- Web Services Standards Support: [Table 1-5](#)
- Other Standards Support: [Table 1-6](#)

**Table 1-4 Java Standards Support**

Standard	Version
J2EE	1.3
JDKs	1.4
J2EE EJB	2.0 and 1.1
J2EE JMS	1.0.2b
J2EE JDBC (with third-party drivers)	2.0
MS SQL jDriver	1.0
Oracle OCI jDriver	1.0 and some 2.0 features (batching)
J2EE JNDI	1.2
OTS/JTA	1.2 and 1.0.1b
J2EE Servlet	2.3 and 2.2

**Table 1-4 Java Standards Support (Continued)**

J2EE JSP	1.2 and 1.1
RMI/IIOP	1.0
JMX	1.0
JavaMail	1.2
JAAS	1.0 Full
J2EE CA	1.0
JCE	1.4
Java RMI	1.0
JAXP	1.1
JAX-RPC	1.0

**Table 1-5 Web Services Standards Support**

<b>Standard</b>	<b>Version</b>
SOAP	1.1 and 1.2
WSDL	1.1
UDDI	v1 and v2
WS-Security	1.0

**Table 1-6 Other Standards Support**

<b>Standard</b>	<b>Version</b>
SSL	v3
X.509	v3
LDAP	v2

**Table 1-6 Other Standards Support**

<b>Standard</b>	<b>Version</b>
TLS	v1
HTTP	1.1
SNMP	--

## Deprecated Features and APIs

For a list of deprecated WebLogic Server classes, see <http://e-docs.bea.com/wls/docs81/javadocs/deprecated-list.html>.

## Third-Party JAR Files

[Table 1-7](#) provides information about the JAR files from third parties that are part of WebLogic Server:

**Table 1-7 Third-Party JAR Files**

JAR File	Description
J2EE, javax and subdirectories	<ul style="list-style-type: none"><li>• com/sun/activation/</li><li>• com/sun/mail</li><li>• corba idl</li><li>• javax/activation</li><li>• javax/connector</li><li>• javax/ejb</li><li>• javax/jms</li><li>• javax/jts</li><li>• javax/mail</li><li>• javax/management</li><li>• javax/net</li><li>• javax/servlet</li><li>• javax/transaction</li><li>• javax/xml/messaging</li><li>• javax/xml/soap</li><li>• javax/xml/rpc</li><li>• jta</li><li>• jts</li></ul>

**Table 1-7 Third-Party JAR Files (Continued)**

JAR File	Description
Libraries	<ul style="list-style-type: none"> <li>• Ant 1.5</li> <li>• Antlr 2.7.1 (<a href="#">MageLang Institute</a>)</li> <li>• Cert-J 2.0.2 from certicom</li> <li>• Certicom SSL 3.1.14</li> <li>• RSA Crypto-J 3.5</li> <li>• Netscape LDAP 3.1</li> <li>• Oracle Thin JDBC driver 9.2.0.3</li> <li>• AdventNet SNMP 3.2 SP1</li> <li>• JavaScript 1.5 from Mozilla</li> <li>• JCom from J-Integra</li> <li>• PointBase 4.3</li> <li>• Octetstring 1.5</li> </ul>
XML	<ul style="list-style-type: none"> <li>• Acumen UDDI</li> <li>• Apache Xerces DOM</li> </ul>

## Other Available Resources

Here are some pointers to useful information related to this release. The hyperlinks require Internet access.

### Fast Track Procedures

High-level procedures to help you quickly deploy an HTML file, JSPs, and servlets, are available at [http://e-docs.bea.com/wls/docs70/quickstart/quick\\_start.html](http://e-docs.bea.com/wls/docs70/quickstart/quick_start.html).

### Examples

Code examples, if installed, are located in the `SAMPLES_HOME\server\src\examples` directory of your WebLogic Server installation, where `SAMPLES_HOME` is the location of all examples for the WebLogic Platform. By default, this location is `c:\bea\weblogic81b\samples`. Examples are also available from the Start menu for Windows users.

## Introduction

For an overview of WebLogic Server features and the J2EE application architecture, see [Introduction to WebLogic Server](#).

## Additional Documentation

The full documentation set for BEA WebLogic Server 8.1, including administration, programming, and reference guides, is provided on the BEA Web site at

<http://e-docs.bea.com/wls/docs81/index.html>.

## Newsgroups

BEA WebLogic Server newsgroups provide community support for BEA products. Information about BEA-related newsgroups can be found at <http://newsgroups.bea.com/> and

<news://newsgroups.bea.com>.

## Dev2Dev Online

The BEA site Dev2Dev Online provides resources to make your e-commerce development easier and faster. To reach Dev2Dev online, go to <http://developer.bea.com/>.

## WebLogic Server 8.1 Features and Changes

# WebLogic Server 8.1 Known Issues

The following sections describe known issues, levels of support, notes, and general problems in WebLogic Server 8.1 and its service packs:

- [“Known Issues in WebLogic Server 8.1 SP5”](#) on page 2-1
- [“Known Issues in WebLogic Server 8.1 SP4”](#) on page 2-2
- [“Known Issues in WebLogic Server 8.1 SP3”](#) on page 2-14
- [“Known Issues in WebLogic Server 8.1 SP2”](#) on page 2-25
- [“Known Issues in WebLogic Server 8.1 SP1”](#) on page 2-46
- [“Known Issues in WebLogic Server 8.1”](#) on page 2-62

## **Known Issues in WebLogic Server 8.1 SP5**

- [“Core WebLogic Server Known Issues”](#) on page 2-2
- [“Security Known Issues”](#) on page 2-2

## Core WebLogic Server Known Issues

---

Change Request Number	Description
CR185820 CR185821	If a client has installed a reverse-proxy server with IP mapping, server IP addresses and domain names may be unavailable.
CR217067	The <code>Low Memory Granularity Level</code> attribute is disabled. WebLogic Server logs a warning message when the average free memory value is smaller than the <code>Low Memory GC Threshold</code> value.
CR236083	Clustered <code>HttpSession</code> replication may trigger a "Received a stale replication request" error when all of the following conditions exist: <ol style="list-style-type: none"><li>1. The cluster is configured with <code>PersistentStoreType</code> set to replicated (and/or replicated if clustered).</li><li>2. The cluster is replicating HTTP sessions for a heavy client load, with several hundred concurrent users.</li><li>3. Some server threads handling client traffic are delayed while waiting for responses from a database.</li></ol>

---

## Security Known Issues

---

Change Request Number	Description
CR182523	<p>Information on how to configure domains to enable inter-domain transactions (that is, all participating domains run on WebLogic Server 9.x, 8.x, 7.x, and 6.x domains or a combination of 9.x, 8.x, 7.x and 6.x) is incorrect in the Administration Console Help.</p> <p><b>Workaround:</b> The information required to configure domains to enable inter-domain transactions is located online. See <a href="#">Configuring Domains for Inter-Domain Transactions</a> in <i>Administration Console Online Help</i>.</p>

---

## Known Issues in WebLogic Server 8.1 SP4

- [“Administration Console Known Issues” on page 2-4](#)

- “Cluster Known Issues” on page 2-5
- “Core WebLogic Server Known Issues” on page 2-6
- “Deployment Known Issues” on page 2-8
- “Documentation Known Issues” on page 2-8
- “Examples Known Issues” on page 2-9
- “Installation Known Issues” on page 2-9
- “JDBC Known Issues” on page 2-10
- “Node Manager Known Issues” on page 2-10
- “Security Known Issues” on page 2-11
- “Transactions Known Issues” on page 2-12
- “Web Services Known Issues” on page 2-13
- “WebLogic Tuxedo Connector (WTC) Known Issues” on page 2-13
- “WebLogic Type 4 JDBC Drivers Known Issues” on page 2-14

## Administration Console Known Issues

---

<b>Change Request Number</b>	<b>Description</b>
CR186747	<p>When viewing the Administration Console in with the language preference set to Japanese or Korean, the links to customize the information displayed on the page (“Customize this view...” in English) do not work.</p> <p><b>Workaround:</b> Select “English” for Language on the Console—Preferences tab in the Administration Console.</p>

---

## Cluster Known Issues

Change Request Number	Description
CR172156	<p>In WebLogic Server 8.1, use of more than about 20 channels in a cluster can result in the formation of multicast header transmissions that exceed the default maximum packet size. The MTUSize attribute in the Server element of config.xml sets the maximum size for packets sent using the associated network card to 1500. Sending packets that exceed the value of MTUSize can result in a java.lang.NegativeArraySizeException, similar to:</p> <pre> java.lang.NegativeArraySizeException at weblogic.common.internal.WLObjectOutputStreamBase2.writeBytes(WLObjectOutputStreamBase2.java:97) at weblogic.cluster.MulticastSender.fragmentAndSend(MulticastSender.java:340) at weblogic.cluster.MulticastSender.send(MulticastSender.java:156) at weblogic.cluster.AttributeManager.sendAttributes(AttributeManager.java:52) at weblogic.cluster.ClusterCommunicationService.resume(ClusterCommunicationService.java:79) at weblogic.t3.srvr.T3Srvr.resume(T3Srvr.java:1042) at weblogic.t3.srvr.T3Srvr.run(T3Srvr.java:359) at weblogic.Server.main(Server.java:32) </pre> <p>When the problem was replicated in a cluster with 22 channels, it was solved by setting MTUSize to 2500.</p> <p><b>Workaround:</b> If you see NegativeArraySizeExceptions in a cluster with a large number of channels, increase incrementally the value of MTUSize until the exception does not occur.</p>

## Core WebLogic Server Known Issues

---

<b>Change Request Number</b>	<b>Description</b>
------------------------------	--------------------

---

CR093104	In earlier service packs, the NT Performance pack caused thread deadlock under some circumstances. The problem was exhibited under light loads during stress tests of a JSP accessing a database. Thread dumps showed the majority of "default" execute threads stuck in "waiting for monitor entry":
----------	---

```
"ExecuteThread: '10' for queue: 'default'" daemon prio=5  
tid=0x273fb548 nid=0xadc waiting for monitor entry  
[0x2810f000..0x2810fdc4]
```

**Workaround:** If you experience such deadlocks, create a `.hotspot_compiler` file in the directory where you invoke the JVM and put the following lines in that file:

```
exclude weblogic/socket/NTSocketMuxer processSockets  
exclude weblogic/socket/NTSocketMuxer trigger  
exclude weblogic/socket/NTSocketMuxer cleanup
```

---

Change Request Number	Description
CR183137	<p>Certain combinations of Java command-line options can cause the Sun JDK to crash with a HotSpot Virtual Machine Error when JSPs are precompiled in production mode (-server) with debug enabled. The symptom is an error like the following:</p> <pre data-bbox="357 517 1197 800"># HotSpot Virtual Machine Error, Internal Error # Please report this error at # http://java.sun.com/cgi-bin/bugreport.cgi # # Java VM: Java HotSpot(TM) Server VM (1.4.2_04-b05 mixed mode) # # Error ID: 53484152454432554E54494D450E435050018D # # Problematic Thread: prio=5 tid=0x2d48f578 nid=0xc10 runnable</pre> <p>In particular, the following combination of options seems to cause this problem:</p> <ul data-bbox="357 855 1197 1025" style="list-style-type: none"> <li>• Domain configured with Sun JDK 1.4.2_04 (not BEA JRockit®)</li> <li>• Server running in production mode (-server option)</li> <li>• Debug enabled</li> <li>• JSP Precompilation enabled (The <code>weblogic.xml</code> file for a Web application specifies <code>precompile=true</code>)</li> </ul> <p>Sun JVM command-line options associated with this issue include:</p> <pre data-bbox="357 1081 974 1164">-server -Xdebug -Xnoagent -Xrunjdwpt:transport=dt_socket,address=8453,server=y,suspend=n -java.compiler=NONE -ea -da:com.bea... -da:javelin... -da:weblogic...</pre> <p><b>Workarounds:</b> There are several ways to avoid this issue (choose one):</p> <ul data-bbox="357 1216 1197 1381" style="list-style-type: none"> <li>• Use the JRockit JVM instead of the Sun JVM.</li> <li>• Edit the server start script to start with <code>-client</code> (not <code>-server</code>) when debugging with the Sun JDK.</li> <li>• Disable debugging (edit <code>setDomainEnv</code> or manually remove debug JVM arguments).</li> <li>• Do not precompile when debugging.</li> </ul>
CR239324 CR239819	<p>If a socket is unregistered and then registered, and if between the two calls the timeout timer never runs, the data structures are getting out of sync which prevents the newly-registered socket from ever timing out.</p> <p>This problem was resolved in 8.1 SP5.</p>

## Deployment Known Issues

Change Request Number	Description
CR210073	<p>When all the modules of an application are pinned (deployed or targeted) to some of the servers in a cluster and no module is targeted to the cluster, a potential deadlock of the Admin RMI threads may cause application deployment to hang.</p> <p><b>Workaround:</b> Increase the RMI pool size by specifying <code>-Dweblogic.management.rmiQueueSize=xx</code> in java command in weblogic startup script. It is recommended that the value of <code>xx</code> be equal to one more than the number of managed servers in the domain. The default value for this pool size is 3.</p>

## Documentation Known Issues

Change Request Number	Description
CR196210	<p>Previous versions of this document and various other sample documents erroneously described using <code>weblogic.management.Admin.getInstance().getAdminMBeanHome()</code> as a way to look up the MBeanHome interface on the Administration Server.</p> <p>However, the <code>weblogic.management.Admin</code> class is not public. Instead of using this non-public class, use JNDI to retrieve MBeanHome. See <i>Determining the Active Domain and Servers in Programming WebLogic Server JMX Services</i>.</p>
n/a	<p>Links in the Administration Console Online Help to the JRockit documentation are to the JRockit 8.1 SP3 documentation instead of the JRockit 1.4.2 documentation. A listing of all JRockit documentation can be found at <a href="http://e-docs.bea.com/more_jrockit.html">http://e-docs.bea.com/more_jrockit.html</a>.</p>

## Examples Known Issues

Change Request Number	Description
CR175804	<p>The <code>iiop.ejb.entity.tuxclient</code> sample does not run successfully.</p> <p>This problem is due to changes in JVM behavior regarding vector marshalling. A BEA Tuxedo® 8.1 rolling patch is expected to be released in the near future to resolve this issue.</p>

## Installation Known Issues

Change Request Number	Description
CR182698	<p>When upgrading to WebLogic Server 8.1 SP3 on Red Hat Enterprise Linux AS or ES 3.0 on Itanium systems, Smart Update hangs and does not complete the installation. (Smart Update works on Red Hat Enterprise Linux 2.1 on Itanium, but not 3.0.)</p> <p><b>Workaround:</b> Do not use Smart Update to upgrade to WebLogic Server 8.1 SP3 on Red Hat Enterprise Linux 3.0. Either use the upgrade installer, or perform a new installation of WebLogic Server 8.1 SP3.</p> <p>For more information about using the upgrade installer, see "<a href="#">Installing Service Packs and Rolling Patches Using a Downloadable Upgrade Installer</a>" in "Installing Service Packs and Rolling Patches" in <i>Installing BEA WebLogic Platform</i>.</p>

## JDBC Known Issues

Change Request Number	Description
CR184298	<p>When using a pooled JDBC connection on a WebLogic Server client, if the JDBC driver classes in the client CLASSPATH are not the exact same version as in the server CLASSPATH, you may see a <code>java.rmi.UnmarshalException</code>. This issue was noted when using a RowSet control in BEA WebLogic Workshop®.</p> <p><b>Workaround:</b> Make sure JDBC driver classes in the client and server CLASSPATHs are the same.</p>
CR190393	<p>CountOfTestFailuresTillFlush is not closing all connections when the stipulated number of test failures has been reached, and CountOfRefreshFailuresTillDisable does not disable the pool after the stipulated number of failures.</p> <p>There is no workaround for this issue in 8.1SP4.</p> <p>This problem was resolved in 8.1 SP5.</p>

## Node Manager Known Issues

Change Request Number	Description
CR203668	<p>When using Managed Servers with the Node Manager, if the Managed Server fails to restart within the time period specified by the Restart Interval attribute and number of attempts specified by the Max Restarts within Interval attribute, the Managed Server will have to be restarted manually.</p>
CR209806	<p>When a machine has been configured in the Admin Console and the listen port for the Node Manager is left unspecified, the Admin Server checks the default port 5555 for the presence of a Node Manager. As a result, the Admin Server may encounter conflict with other processes running on port 5555 even when the Node Manager is not deployed.</p>

## Security Known Issues

<b>Change Request Number</b>	<b>Description</b>
CR187784	The context path and uri arguments for the URLResource are stored in lowercase on Windows and mixed case on UNIX. As a result, if you define roles and policies in a domain on one operating system, export them, and then import them to a domain on the other operating system, the roles and policies may not work correctly. The end result is that webapp pages that you protect on one operating system may not be protected when the roles and policies are imported to a domain on the other operating system.
CR206178	Using WLS8.1-SP3 with JDK 1.4.2 and making an outbound ssl connection causes the following error: "java.lang.IllegalStateException: Cipher not initialized". The problem is resolved when the customer copied all the jars from the /jre/lib/ext folder of JDK 1.4.1 into the /jre/lib/ext of JDK 1.4.2.

## Transactions Known Issues

Change Request Number	Description
CR201308	<p>If your application is using XA transactions with the Oracle thin driver 10.1.0.3, you must grant execute permission on the dbms_system table to the database user used to connect to the database, along with other database preparations for XA:</p> <ol style="list-style-type: none"> <li>Execute the following command: @xaview.sql</li> <li>Grant the following permissions:</li> </ol> <pre>grant select on v\$xsatrans\$ to public (or &lt;user&gt;); grant select on pending_trans\$ to public; grant select on dba_2pc_pending to public; grant select on dba_pending_transactions to public;</pre> <p><b>(when using the Oracle Thin driver 10.1.0.3 or later)</b></p> <pre>grant execute on dbms_system to &lt;user&gt;;</pre> <p>If the above steps are not performed on the database server, normal XA database queries and updates may work fine. However, when the Weblogic Server Transaction Manager performs recovery on a re-boot after a crash, recover for the Oracle resource will fail with XAER_RMERR. Crash recovery is a standard operation for an XA resource.</p> <p><b>Note:</b> Weblogic Server 8.1SP4 ships with the Oracle Thin driver 10.1.0.2, so granting permission on the dbms_system is required only if you use a different version of the driver.</p> <p>See “<a href="#">Using the Oracle Thin/XA Driver</a>” in <i>Programming WebLogic JTA</i> for more information about using the Oracle Thin/XA driver with WebLogic Server.</p>
CR220662	<p>Under certain failure conditions following a database server or database network crash, pending transaction branches on failed Oracle Real Application Clusters (RAC) instances may not be properly resolved. Use one of the following methods to correct this problem:</p> <ul style="list-style-type: none"> <li>To prevent data loss, restart the coordinating managed server</li> <li>If data loss is acceptable, forcibly roll back the transaction in the Oracle database</li> </ul>

## Web Services Known Issues

Change Request Number	Description
CR107595	<p>SSL does not work for J2ME clients on WebLogic Server 8.1.</p> <p>Certicom SSL libraries require additional features that are not supported by J2ME. Therefore, SSL is not supported for J2ME clients on WebLogic Server 8.1.</p>

## WebLogic Tuxedo Connector (WTC) Known Issues

Change Request Number	Description
CR185475	<p>WTC cannot have multiple connections within a single transaction.</p> <p>If multiple TuxedoConnections are instantiated within a single transaction, only the first TuxedoConnection instance is enlisted in the transaction.</p>
CR179956	<p>WTC imported service Resource Name must match Remote Name.</p> <p>WTC imported service Resource Name and Remote Service Name cannot differ. When the imported service Mbean is updated (to give a different resource name), the WTCService cache is not updated.</p> <p><b>Workaround:</b> Reboot the WebLogic Server instance.</p>
CR127660	<p>View classes are not set on a per connection basis.</p> <p>A shared BEA WebLogic Tuxedo Connector™ hash table can cause unexpected behavior to the server if two applications point to the same VIEW name with different definitions. There should be a hash table for the view classes on the connection as well as for the Resource section.</p> <p><b>Workaround:</b> Ensure that all VIEW classes defined across all your WebLogic Workshop applications are consistent, meaning that you have the same VIEW name representing the same VIEW class.</p>

## WebLogic Type 4 JDBC Drivers Known Issues

Change Request Number	Description
CR202674	<p>The WebLogic Type 4 JDBC driver for DB2 is not included in the WebLogic Server 8.1 SP4 release. It was removed because DB2 support is not yet available for WebLogic Server 8.1 SP4. For customers that require DB2 support, BEA recommends that you use WebLogic Server 8.1 SP3 until DB2 support is available for WebLogic Server 8.1 SP4.</p>
--	<p><b>Oracle Driver</b></p> <p>The WebLogic JDBC Oracle driver now describes columns defined as FLOAT or FLOAT(n) as a DOUBLE SQL type. The driver previously described these columns as a FLOAT SQL type. Both the DOUBLE type and the FLOAT type represent a double precision floating point number.</p>
--	<p><b>SQL Server Driver</b></p> <p>The SendStringParameterAsUnicode connection option is set to true by default. Setting this option to true can negatively impact performance if an application does string comparison in SQL and the data stored in the database is not Unicode. On the other hand, setting this option to false can cause data corruption or unexpected comparison results if the data is not stored in the default database character set. Microsoft SQL Server allows you to define columns in a table that use a character set other than the default database character set. When sending parameter data to the server, the driver does not know what the character set is for the column that is being compared or modified. Sending parameter data as Unicode will always work, but is subject to the conversion and performance issues mentioned in the Microsoft document: <a href="http://support.microsoft.com/default.aspx?scid=kb;en-us;Q271566">http://support.microsoft.com/default.aspx?scid=kb;en-us;Q271566</a>.</p>
CR183190	<p>Under certain statement failure conditions, cached statements are leaked without being closed, which can lead to DBMS resource problems. When using a DB2 database, you may see the following error:</p> <pre data-bbox="292 1269 1022 1321">"java.sql.SQLException: [BEA][DB2 JDBC Driver]No more available statements.</pre> <p>Please recreate your package with a larger dynamicSections value."</p> <p>There is no workaround for this issue in 8.1SP4.</p> <p>This problem was resolved in 8.1 SP5.</p>

## Known Issues in WebLogic Server 8.1 SP3

The following sections describe known issues in WebLogic Server 8.1 SP3:

- [“Administration Console Known Issues” on page 2-27](#)
- [“Cluster Known Issues” on page 2-17](#)
- [“Core WebLogic Server Known Issues” on page 2-18](#)
- [“Documentation Known Issues” on page 2-21](#)
- [“Examples Known Issues” on page 2-21](#)
- [“Installation Known Issues” on page 2-21](#)
- [“JDBC Known Issues” on page 2-22](#)
- [“RMI/IIOP Known Issues” on page 2-23](#)
- [“Security Known Issues” on page 2-23](#)
- [“Transactions Known Issues” on page 2-24](#)
- [“Web Services Known Issues” on page 2-25](#)
- [“WebLogic Tuxedo Connector \(WTC\) Known Issues” on page 2-25](#)

## Administration Console Known Issues

---

<b>Change Request Number</b>	<b>Description</b>
CR186747	<p>When viewing the Administration Console in with the language preference set to Japanese or Korean, the links to customize the information displayed on the page (“Customize this view...” in English) do not work.</p> <p><b>Workaround:</b> Select “English” for Language on the Console—Preferences tab in the Administration Console.</p>

---

## Cluster Known Issues

Change Request Number	Description
CR172156	<p>In WebLogic Server 8.1, use of more than about 20 channels in a cluster can result in the formation of multicast header transmissions that exceed the default maximum packet size. The MTUSize attribute in the Server element of config.xml sets the maximum size for packets sent using the associated network card to 1500. Sending packets that exceed the value of MTUSize can result in a java.lang.NegativeArraySizeException, similar to:</p> <pre> java.lang.NegativeArraySizeException at weblogic.common.internal.WLObjectOutputStreamBase2.writeBytes(WLObjectOutputStreamBase2.java:97) at weblogic.cluster.MulticastSender.fragmentAndSend(MulticastSender.java:340) at weblogic.cluster.MulticastSender.send(MulticastSender.java:156) at weblogic.cluster.AttributeManager.sendAttributes(AttributeManager.java:52) at weblogic.cluster.ClusterCommunicationService.resume(ClusterCommunicationService.java:79) at weblogic.t3.srvr.T3Srvr.resume(T3Srvr.java:1042) at weblogic.t3.srvr.T3Srvr.run(T3Srvr.java:359) at weblogic.Server.main(Server.java:32) </pre> <p>When the problem was replicated in a cluster with 22 channels, it was solved by setting MTUSize to 2500.</p> <p><b>Workaround:</b> If you see NegativeArraySizeExceptions in a cluster with a large number of channels, increase incrementally the value of MTUSize until the exception does not occur.</p>

## Core WebLogic Server Known Issues

---

<b>Change Request Number</b>	<b>Description</b>
------------------------------	--------------------

---

CR093104	In earlier service packs, the NT Performance pack caused thread deadlock under some circumstances. The problem was exhibited under light loads during stress tests of a JSP accessing a database. Thread dumps showed the majority of "default" execute threads stuck in "waiting for monitor entry":
----------	---

```
"ExecuteThread: '10' for queue: 'default'" daemon prio=5  
tid=0x273fb548 nid=0xadc waiting for monitor entry  
[0x2810f000..0x2810fdc4]
```

**Workaround:** If you experience such deadlocks, create a `.hotspot_compiler` file in the directory where you invoke the JVM and put the following line in that file:

```
exclude weblogic/socket/NTSocketMuxer processSockets
```

---

Change Request Number	Description
CR183137	<p>Certain combinations of Java command-line options can cause the Sun JDK to crash with a HotSpot Virtual Machine Error when JSPs are precompiled in production mode (<code>-server</code>) with debug enabled. The symptom is an error like the following:</p> <pre data-bbox="357 517 1206 798"># HotSpot Virtual Machine Error, Internal Error # Please report this error at # http://java.sun.com/cgi-bin/bugreport.cgi # # Java VM: Java HotSpot(TM) Server VM (1.4.2_04-b05 mixed mode) # # Error ID: 53484152454432554E54494D450E435050018D # # Problematic Thread: prio=5 tid=0x2d48f578 nid=0xc10 runnable</pre> <p>In particular, the following combination of options seems to cause this problem:</p> <ul data-bbox="357 852 1206 1017" style="list-style-type: none"> <li>• Domain configured with Sun JDK 1.4.2_04 (not JRockit)</li> <li>• Server running in production mode (<code>-server</code> option)</li> <li>• Debug enabled</li> <li>• JSP Precompilation enabled (The <code>weblogic.xml</code> file for a Web application specifies <code>precompile=true</code>)</li> </ul> <p>Sun JVM command-line options associated with this issue include:</p> <pre data-bbox="357 1072 974 1156">-server -Xdebug -Xnoagent -Xrunjdwp:transport=dt_socket,address=8453,server=y,suspend=n -java.compiler=NONE -ea -da:com.bea... -da:javelin... -da:weblogic...</pre> <p><b>Workarounds:</b> There are several ways to avoid this issue (choose one):</p> <ul data-bbox="357 1211 1206 1373" style="list-style-type: none"> <li>• Use the JRockit JVM instead of the Sun JVM.</li> <li>• Edit the server start script to start with <code>-client</code> (not <code>-server</code>) when debugging with the Sun JDK.</li> <li>• Disable debugging (edit <code>setDomainEnv</code> or manually remove debug JVM arguments).</li> <li>• Do not precompile when debugging.</li> </ul>

---

<b>Change Request Number</b>	<b>Description</b>
CR240499	<p>In HP-UX, the thread runs with the default thread-priority assigned by the operating system even when a different thread priority is set from the Administration Console.</p> <p><b>Workaround:</b></p> <ol style="list-style-type: none"><li>1. As a root user, run the following command <code>setprivgrp &lt;user-group&gt; RTSCHED</code> where, <i>user-group</i> is the unix-group.</li><li>2. Add the following java command line option to start WebLogic script <code>-XX:SchedulerPriorityRange=SCHED_RTprio</code></li></ol>

---

## Documentation Known Issues

Change Request Number	Description
n/a	Links in the Administration Console Online Help to the JRockit documentation are to the JRockit 8.1 SP3 documentation instead of the JRockit 1.4.2 documentation. A listing of all JRockit documentation can be found at <a href="http://e-docs.bea.com/more_jrockit.html">http://e-docs.bea.com/more_jrockit.html</a> .

## Examples Known Issues

Change Request Number	Description
CR175804	<p>The <code>iiop.ejb.entity.tuxclient</code> sample does not run successfully.</p> <p>This problem is due to changes in JVM behavior regarding vector marshalling. A Tuxedo 8.1 rolling patch is expected to be released in the near future to resolve this issue.</p>

## Installation Known Issues

Change Request Number	Description
CR182698	<p>When upgrading to WebLogic Server 8.1 SP3 on Red Hat Enterprise Linux AS or ES 3.0 on Itanium systems, Smart Update hangs and does not complete the installation. (Smart Update works on Red Hat Enterprise Linux 2.1 on Itanium, but not 3.0.)</p> <p><b>Workaround:</b> Do not use Smart Update to upgrade to WebLogic Server 8.1 SP3 on Red Hat Enterprise Linux 3.0. Either use the upgrade installer, or perform a new installation of WebLogic Server 8.1 SP3.</p> <p>For more information about using the upgrade installer, see "<a href="#">Installing Service Packs and Rolling Patches Using a Downloadable Upgrade Installer</a>" in "Installing Service Packs and Rolling Patches" in <i>Installing BEA WebLogic Platform</i>.</p>

## JDBC Known Issues

Change Request Number	Description
CR179600	<p>Under certain statement failure conditions, cached statements are leaked without being closed, which can lead to DBMS resource problems. When using an Oracle database, this can result in out-of-cursor failures.</p> <p>A patch is available for this issue. To get the patch, contact BEA Customer Support and reference CR179600.</p> <p>This problem was resolved in 8.1 SP4.</p>
CR183190	<p>Under certain statement failure conditions, cached statements are leaked without being closed, which can lead to DBMS resource problems. When using a DB2 database, you may see the following error:</p> <pre>"java.sql.SQLException: [BEA][DB2 JDBC Driver]No more available statements.</pre> <p>Please recreate your package with a larger <code>dynamicSections</code> value."</p> <p>There is no workaround for this issue in 8.1SP3.</p> <p>This problem was resolved in 8.1 SP5.</p>
CR184298	<p>When using a pooled JDBC connection on a WebLogic Server client, if the JDBC driver classes in the client <code>CLASSPATH</code> are not the exact same version as in the server <code>CLASSPATH</code>, you may see a <code>java.rmi.UnmarshalException</code>. This issue was noted when using a <code>RowSet</code> control in WebLogic Workshop.</p> <p><b>Workaround:</b> Make sure JDBC driver classes in the client and server <code>CLASSPATHs</code> are the same.</p>
CR190393	<p><code>CountOfTestFailuresTillFlush</code> is not closing all connections when the stipulated number of test failures has been reached, and <code>CountOfRefreshFailuresTillDisable</code> does not disable the pool after the stipulated number of failures.</p> <p>There is no workaround for this issue in 8.1SP3.</p> <p>This problem was resolved in 8.1 SP5.</p>

## RMI/IIOP Known Issues

Change Request Number	Description
CR124596 CR192252	<p>The enhancement described in CR124596 was not fully implemented in WebLogic Server 8.1SP3. The optional enhancement to the BEA ORB forces reconnection when bootstrapping and allows hardware load-balancers to correctly balance connection attempts.</p> <p>A patch for WebLogic Server 8.1SP3 is now available. To get the patch, contact BEA Customer Support and reference the patch for CR192252.</p> <p>This problem was resolved in 8.1 SP4 for CR192252.</p>

## Security Known Issues

Change Request Number	Description
CR184421	<p>When importing security data into a new security realm, the user <code>weblogic</code> loses membership in all groups except <code>Administrators</code>. After importing the data, you must manually add <code>weblogic</code> to any required groups.</p>
CR187784	<p>The context path and uri arguments for the <code>URLResource</code> are stored in lowercase on Windows and mixed case on UNIX. As a result, if you define roles and policies in a domain on one operating system, export them, and then import them to a domain on the other operating system, the roles and policies may not work correctly. The end result is that webapp pages that you protect on one operating system may not be protected when the roles and policies are imported to a domain on the other operating system.</p>

## Transactions Known Issues

Change Request Number	Description
CR193458	<p>When using DataDirect Connect for BEA WebLogic JDBC Drivers™ version 3.4 with a DB2 database, you may see XAER_RMERR errors for applications that previously ran without errors using the 3.3 version of the drivers. The problem is caused by implicit closing of result sets, a feature added to the drivers to improve performance.</p> <p><b>Workaround:</b> DataDirect added a connection property to control implicit result set closing: <code>AllowImplicitResultSetCloseForXA</code>. The feature is enabled by default. To disable implicit result set closings, set <code>AllowImplicitResultSetCloseForXA=false</code>. To make this setting for a JDBC connection pool, add it to the Properties for the connection pool on the JDBC Connection Pool <del>→</del>Configuration <del>→</del>General tab in the Administration Console. The resulting change in the <code>config.xml</code> file resembles the following <code>config.xml</code> excerpt:</p> <pre data-bbox="391 874 1072 1220"> &lt;JDBCConnectionPool DriverName="com.ddtek.jdbcx.db2.DB2DataSource" Name="myPool" Properties="user=scott; portNumber=50000; databaseName=SAMPLE; serverName=dbserver1.yourco.COM; batchPerformanceWorkaround=true; <b>AllowImplicitResultSetCloseForXA=false</b> ... URL="jdbc:datadirect:db2://dbserver1:50000;DatabaseName=SAMPLE" /&gt; </pre> <p><b>Note:</b> This issue applies to the DataDirect Connect for JDBC version 3.4 drivers for DB2 only. The problem has not been observed with drivers for other database management systems or previous releases of the DB2 drivers. <code>AllowImplicitResultSetCloseForXA</code> is ignored for these other drivers.</p>

## Web Services Known Issues

Change Request Number	Description
CR107595	<p>SSL does not work for J2ME clients on WebLogic Server 8.1.</p> <p>Certicom SSL libraries require additional features that are not supported by J2ME. Therefore, SSL is not supported for J2ME clients on WebLogic Server 8.1.</p>

## WebLogic Tuxedo Connector (WTC) Known Issues

Change Request Number	Description
CR185475	<p>WTC cannot have multiple connections within a single transaction.</p> <p>If multiple TuxedoConnections are instantiated within a single transaction, only the first TuxedoConnection instance is enlisted in the transaction.</p>
CR179956	<p>WTC imported service Resource Name must match Remote Name.</p> <p>WTC imported service Resource Name and Remote Service Name cannot differ. When the imported service Mbean is updated (to give a different resource name), the WTCService cache is not updated.</p> <p><b>Workaround:</b> Reboot the WebLogic Server instance.</p>
CR127660	<p>View classes are not set on a per connection basis.</p> <p>A shared WebLogic Tuxedo Connector hash table can cause unexpected behavior to the server if two applications point to the same VIEW name with different definitions. There should be a hash table for the view classes on the connection as well as for the Resource section.</p> <p><b>Workaround:</b> Ensure that all VIEW classes defined across all your WebLogic Workshop applications are consistent, meaning that you have the same VIEW name representing the same VIEW class.</p>

## Known Issues in WebLogic Server 8.1 SP2

The following sections describe known issues in WebLogic Server 8.1 SP2:

- “Patch Available for Sun Java 2 1.4.2 SDK, Oracle 10g Driver, and SQL Server, Sybase or DB2 Database Users” on page 2-26
- “Administration Console Known Issues” on page 2-27
- “Configuration Wizard Known Issues” on page 2-27
- “Connector Known Issues” on page 2-28
- “CORE Known Issues” on page 2-30
- “EJB Known Issues” on page 2-31
- “Examples Known Issues” on page 2-32
- “General Known Issues” on page 2-32
- “Installation Known Issues” on page 2-33
- “JMS Known Issues” on page 2-34
- “JVM Known Issues” on page 2-35
- “Security Known Issues” on page 2-37
- “Servlet & JSP Known Issues” on page 2-39
- “Transactions Known Issues” on page 2-39
- “WebLogic JDBC Type 4 Drivers Known Issues” on page 2-40
- “Web Services Known Issues” on page 2-44

## Patch Available for Sun Java 2 1.4.2 SDK, Oracle 10g Driver, and SQL Server, Sybase or DB2 Database Users

If you are using WebLogic Server or BEA WebLogic Platform™ 8.1 SP2 with the Sun Java 2 1.4.2 SDK, or the Oracle 10g driver, or a SQL Server, Sybase or DB2 database, you may need to install the WebLogic Platform 8.1 SP2 SDK1.4.2/Oracle10gdriver/Database patch. This patch, and a description of the specific configurations that require it, are available at the following dev2dev Web site:

[http://dev2dev.bea.com/products/wlplatform81/patch/wlplat81sp2\\_patch.jsp](http://dev2dev.bea.com/products/wlplatform81/patch/wlplat81sp2_patch.jsp)

## Administration Console Known Issues

Change Request Number	Description
CR125552	<p>When using the WebLogic Server 8.1 Administration Console on machines that have JDK version 1.4.1_02 installed, the navigation applet fails to load and the system freezes.</p> <p><b>Workaround:</b> Uninstall JDK version 1.4.1_02 from the system and reinstall version JDK version 1.4.2_01 or later.</p>
CR129594	<p>The WebLogic Server Administration Console's JDBC Connection Pool Configuration tab does not have an option to rollback local transactions upon closing a connection.</p> <p><b>Workaround:</b> Set this option by editing the <code>config.xml</code> JDBCConnectionPool attribute <a href="#">RollbackLocalTxUponConnClose</a>, or via the Configuration Wizard.</p>
CR132229	<p>The separators for multiple addresses in BEA Jolt™ and WLEC ConnectionPools are incorrectly identified as semicolons.</p> <p><b>Workaround:</b> Use commas to separate multiple addresses when configuring Jolt and WLEC Primary and Secondary Address fields.</p>

## Configuration Wizard Known Issues

Change Request Number	Description
CR125514	<p>When running the <code>config_builder.sh</code> script from the <code>BEA_HOME/wls81/common/bin</code> directory on a Solaris platform, the Configuration Template Builder does not start. The message "Unable to instantiate GUI, defaulting to console mode" is displayed, but rather than starting up in Console mode, a command prompt is displayed. The Configuration Template Builder is not supported in console mode—a graphics environment is required.</p> <p><b>Workaround:</b> Run the Configuration Wizard using xWindows.</p> <p><b>Note:</b> The error message will be changed in the next service pack to more accurately communicate this limitation.</p>

## Connector Known Issues

Change Request Number	Description
n/a	<p>When a connection request is made, WebLogic Server returns to the client (via the resource adapter) a Proxy object that wraps the connection object. WebLogic Server uses this proxy to provide features that assist applications using the WebLogic Server implementation of the J2EE Connector Architecture. These features include (1) connection leak detection capabilities and (2) late XAResource enlistment when a connection request is made before starting a global transaction that uses that connection.</p> <p>If the connection object returned from a connection request is cast as a <code>Connection</code> class, a <code>ClassCastException</code> can occur. This exception is caused by either (1) the resource adapter doing the cast or (2) the client doing the cast during a connection request.</p> <p>In WebLogic Server 8.1 SP2, an attempt is made to detect the <code>ClassCastException</code> caused by the resource adapter case (1) above. If the server detects that this cast is failing, it turns off the proxy wrapper feature and proceeds by returning the connection object during a connection request (unwrapped). The server logs a warning message to indicate that the proxy wrapper has been turned off. When this type of cast failure occurs, connection leak detection and late XAResource enlistment features are also turned off (but currently no indication of this is given in the Console monitoring).</p> <p>WebLogic Server attempts to detect the <code>ClassCastException</code> by acting as a client using container-managed security. Doing so requires the resource adapter to be deployed with security credentials defined.</p> <p>If the client is doing the cast and getting a <code>ClassCastException</code>, the customer (client) code can be modified as follows:</p> <p><b>Workaround:</b> If the client casts the connection object to <code>MyConnection</code>, instead of <code>MyConnection</code> being a class that implements the resource adapter's <code>Connection</code> interface, modify it to be an interface that extends <code>Connection</code>. Implement a <code>MyConnectionImpl</code> class that implements the <code>MyConnection</code> interface.</p>

<b>Change Request Number</b>	<b>Description</b>
CR100048	If two adapters are configured to use the same Enterprise Information System (EIS) with the same Resource Manager, then using the two adapters in the same transaction can result in deadlocks or long delays followed by a transaction failure. This is because most Resource Managers, including Oracle's, do not allow adapters to share an EIS.
CR127127	<p>The IIOP thin-client is not supported on AIX (or other platforms with the IBM VM) due to dependencies on the Sun VM.</p> <p><b>Note:</b> Only clients are affected by this issue. A thin-client running on another platform will work just fine against WebLogic Server running on AIX.</p>

## CORE Known Issues

---

<b>Change Request Number</b>	<b>Description</b>
CR106616	After several operations on the WebLogic Server Administration Console, Netscape Version 4.79 running on Linux AS2.1 and HP-UX shuts down with a bus error, sometimes with a core dump.

---

## EJB Known Issues

Change Request Number	Description
CR112393	<p>An EJB JAR is referencing another utility JAR through the Manifest classpath. When you compile an EJB using the <code>weblogic.appc</code> utility, classes included in the referenced JAR are not found. The <code>appc</code> compiler explodes the source JAR in a temporary directory and the classfinder looks for the dependent JAR in that temporary directory; however, the classes are not copied into the temporary directory.</p> <p><b>Workaround:</b></p> <p>This issue only exists for stand-alone modules that are not part of an EAR file. BEA recommends that you package the EJB JAR file and utility JAR file together in an EAR file. This will ensure they are always tied together. If you choose not to do this, BEA recommends you use the <code>weblogic.ejbcb</code> utility rather than <code>weblogic.appc</code>.</p> <p><b>Note:</b> The <code>weblogic.ejbcb</code> utility is a deprecated utility and will not be available in future releases.</p>
CR126613	<p>An MDB running in a transactional context interacts with an external system (such as an LDAP server) and message processing takes longer than transaction timeout value. The MDB throws the following exception and message processing stops:</p> <pre data-bbox="455 1065 1137 1459">weblogic.utils.AssertionError: ***** ASSERTION FAILED *****[ Left-over JTA transaction found on MDB listener thread ] at weblogic.utils.Debug.assertion(Debug.java:57) at weblogic.ejb20.internal.MDListener.onMessage(MDList stener.java:245) at weblogic.jms.client.JMSSession.onMessage(JMSSessi on.java:2596) at weblogic.jms.client.JMSSession.execute(JMSSession .java:2516) at weblogic.kernel.ExecuteThread.execute(ExecuteThre ad.java:197) at weblogic.kernel.ExecuteThread.run(ExecuteThread.j ava:170)</pre> <p><b>Workaround:</b> Increase the value of the transaction timeout to make it high enough.</p>

## Examples Known Issues

Change Request Number	Description
CR104550	<p>The <code>config.xml</code> file for the samples banking application domain contains an incorrect name for the EJB. With this error, the application cannot find the EJB.</p> <pre>&lt;Application Deployed="true" Name="app_banking"   Path=". \config\migrationdomain\applications"   StagedTargets="migrationserver"&gt; &lt;EJBComponent Name="app_banking"   Targets="migrationserver"   URI="app_banking.jar" /&gt; &lt;/Application&gt;</pre> <p>It should contain:</p> <pre>&lt;Application Deployed="true"   Name="containerManaged"   Path=". \config\migrationdomain\applications"   StagedTargets="migrationserver"&gt; &lt;EJBComponent Name="containerManaged"   Targets="migrationserver"   URI="app_banking.jar" /&gt; &lt;/Application&gt;</pre> <p><b>Workaround:</b> Undeploy the EJB, and redeploy it with the correct name, <code>containerManaged</code>.</p>

## General Known Issues

Change Request Number	Description
CR128912	<p>There is a known issue in HP-UX SDK 1.4.1.05 character set API and as a result, <code>Charset.availableCharsets()</code> always fails. This problem manifests itself in WebLogic Server SP2 using JVM 1.4.1.05.</p> <p><b>Workaround:</b> Download and use HP's 1.4.1.06 JVM, available as part of the JDK 1.4.1.06 at <a href="http://www.hp.com/products1/unix/java/">http://www.hp.com/products1/unix/java/</a>.</p>

## Installation Known Issues

---

<b>Change Request Number</b>	<b>Description</b>
CR127350	As part of the upgrade procedure, the WebLogic Platform 8.1 SP2 installation program automatically overwrites any files, including applications, that you have created in <i>WL_HOME</i> . <b>Workaround:</b> Before installing SP2, make backup copies of any files or applications that you have created in <i>WL_HOME</i> .

---

## JMS Known Issues

Change Request Number	Description
CR125979	<p>Customers who set up a JMS bridge within a cluster may find that when starting the bridge, the following exception is thrown from the Managed Servers, and that the JMS message cannot be sent to the intended destination.</p> <pre data-bbox="391 586 1079 1072"> &lt;Oct 16, 2003 3:12:49 PM PDT&gt; &lt;Notice&gt; &lt;WebLogicServer&gt; &lt;BEA-000360&gt; &lt;Server started in RUNNING mode&gt; &lt;Oct 16, 2003 3:23:17 PM PDT&gt; &lt;Error&gt; &lt;JMS&gt; &lt;BEA-040368&gt; &lt;The following exception has occurred: java.lang.NullPointerException at weblogic.jms.bridge.internal.MessagingBridge.startInternal(MessagingBridge.java:519) at weblogic.jms.bridge.internal.MessagingBridge.execute(MessagingBridge.java:956) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:197) at weblogic.kernel.Kernel.execute(Kernel.java:336) at weblogic.kernel.Kernel.execute(Kernel.java:321) at ... </pre> <p><b>Workaround:</b></p> <p>Deploy the adapter to individual servers instead of to the entire cluster. The adapter must be deployed on the same server to which the bridge is targeted. For instructions, see <a href="http://e-docs.bea.com/wls/docs81/ConsoleHelp/messaging_bridge.html#1109910">http://e-docs.bea.com/wls/docs81/ConsoleHelp/messaging_bridge.html#1109910</a>.</p> <p>This problem was resolved in 8.1 SP3.</p>
CR128596	<p>In some cases, messages remain in the source queue of the JMS bridge in the sending-side cluster after restarting a Managed Server on the receiving side cluster.</p> <p><b>Workaround:</b> Apply the patch for CR128596 to all WebLogic Server instances.</p> <p>This problem was resolved in 8.1 SP3.</p>

## JVM Known Issues

Change Request Number	Description
CR131717	<p>Due to a bug in the Sun JDK version 1.4.2_02, a server ExecuteThread freezes or a client JVM hangs when using JDK version 1.4.2_02 with Japanese encodings ISO-2022-JP and ISO2022JP. Customers not using Japanese encodings will not be affected by this bug.</p> <p><b>Workaround:</b> This bug will be fixed in JDK version 1.4.2_04 (available February 12, 2004), as documented at <a href="http://developer.java.sun.com/developer/bugParade/bugs/4879522.html">http://developer.java.sun.com/developer/bugParade/bugs/4879522.html</a>. In the meantime, Japanese customers can use JDK version 1.4.1.</p>
CR132292	<p>A change in behavior of <code>javac</code> in JDK version 1.4.2 causes tag handler classes written as inner classes to generate compilation errors. The code generated contains a “<code>ş</code>” character in the type name, which is not allowed in JDK version 1.4.2.</p> <p><b>Workaround:</b> Declare tag handler classes as public classes instead of inner classes. Or, contact support to obtain a patch for WebLogic Server 8.1 SP2.</p>
CR135783	<p>Calling <code>WebLogicMBeanMaker</code> to package authentication provider MBeans yields different results between JDK version 1.4.1_05 and 1.4.2_03.</p> <p>With JDK 1.4.1_05, when calling <code>WebLogicMBeanMaker</code> with a source folder that contains no class files, <code>WebLogicMBeanMaker</code> silently ignores the error and creates an MBean JAR file.</p> <p>With JDK 1.4.2_03, <code>WebLogicMBeanMaker</code> fails if there are no class files in the source path.</p>
CR136167	<p>The Administration Console cannot display the server log in a Japanese locale because of a problem in JDK 1.4.2_03 and earlier JDK 1.4.2 releases.</p> <p><b>Workaround:</b> Use the Sun JDK1.4.2_04 or JRockit JDK1.4.2_04.</p>

## Plug-In Known Issues

---

Change Request Number	Description
CR130088	<p>The IIS proxy plugin performs very slowly when the IIS HTTP Keep-Alives Enabled check box is checked. The symptom is a 20-40x slowdown compared to normal performance. It is also possible to receive <code>PROTOCOL_ERROR</code> messages under certain keepalive configurations.</p> <p><b>Workaround:</b> If you see slow performance or <code>PROTOCOL_ERROR</code> messages with the IIS proxy plugin, switch to the following IIS/proxy plugin keepalive configuration:</p> <ol style="list-style-type: none"><li>1. Configure IIS so the 'HTTP Keep-Alives Enabled' check box is NOT checked.  Open the Internet Services Manager tool so you can configure IIS. In the left-hand pane, right-click on the Web site item which proxies activity for your server. Select 'Properties' Select the 'Web Site' tab. Ensure the 'HTTP Keep-Alives Enabled' check box is NOT checked. Click Apply, then OK.</li><li>2. Configure the proxy plugin so <code>KeepAliveEnabled</code> is false in the <code>iis_proxy.ini</code> file.  Edit the <code>iis_proxy.ini</code> file in a text editor, and set <code>KeepAliveEnabled=false</code>.</li><li>3. Restart IIS.  In the left-hand pane, right-click on the IIS server name, and select Restart IIS...  Select Restart Internet Services on &lt;IIS servername&gt;. Click OK.</li></ol> <p>Once IIS restarts, the proxy performance should improve to normal levels. This problem was resolved in 8.1 SP3.</p>

---

## Security Known Issues

---

<b>Change Request Number</b>	<b>Description</b>
CR064593	<p>In the new security framework in WebLogic Server, SSL does not cache the sessions; instead SSL initiates a new handshake for each request causing performance issues and out of memory exceptions.</p> <p>In WebLogic 7.0 and 8.1 SP2, WebLogic Server added the <code>weblogic.security.SSL.sessionCache.ttl</code> which maintains the time to live for the SSL session. The default value is 90000 milliseconds (90 seconds) which means if a client accesses the server again (via the same session ID) within 90 seconds, WebLogic Server will use the existing SSL session. You can change this value by setting <code>-Dweblogic.security.SSL.sessionCache.ttl</code> in the server startup script.</p>

---

Change Request Number	Description
CR126610	<p>Customers using a BEA WebLogic Express™ license are not able to configure global security roles or global security policies using the WebLogic Server Administration Console. This behavior is as designed, and this limitation will be updated in the documentation.</p>
CR130185	<p>The JDK <code>cacerts</code> file and the WebLogic Server <code>cacerts</code> file are keystore files that contain out-of-the-box trusted Certificate Authority certificates. These trusted CA certs are used in certificate chain verification when using SSL.</p> <p>All certificates, including trusted CAs, have <code>NotBefore</code> and <code>NotAfter</code> dates contained within them. A certificate cannot be validated prior to or after these validity dates, respectively.</p> <p>In this service pack, WebLogic Server checks each trusted CA certificate in the JDK and WebLogic Server <code>cacerts</code> files for the <code>NotAfter</code> date. If the trusted CA is due to expire (that is, exceed the <code>NotAfter</code> date) within 30 days, a warning message is output about this impending expiration. If a trusted CA has already expired, with one exception, a warning message is output declaring the trusted CAs as having exceeded the <code>NotAfter</code> date.</p> <p>There are two trusted CAs in the JDK <code>cacerts</code> keystore file due to expire approximately at the release date of this service pack. Thus, you may see one or both of these warning messages output while booting WebLogic Server.</p> <p><b>Workaround:</b> If you use either of these two trusted CAs in your server certificate chain, you should obtain a new trusted CA or certificate from your Certificate Authority of choice (Verisign, Certicom, and so on). For more information about expiring and renewing certificates, contact your preferred Certificate Authority.</p> <p>The current JDK <code>cacerts</code> file contains at least one certificate from Verisign that has expired (the Class 4 Certificate Authority, alias <code>verisignclass4ca</code>). No certificate expired message is output when this trusted CA is loaded.</p> <p>For more information, see Sun Alert 57436 at <a href="http://sunsolve.sun.com/pub-cgi/retrieve.pl?doc=fsalert%2F57436">http://sunsolve.sun.com/pub-cgi/retrieve.pl?doc=fsalert%2F57436</a>.</p>

## Servlet & JSP Known Issues

Change Request Number	Description
CR125201	<p>A <code>ClassNotFoundException</code> occurs when migrating sessions to WebLogic Server 8.1 SP2 using secure URLs.</p> <p><b>Note:</b> WebLogic Server does not support two servers in different releases interacting in this way.</p>
CR126809	<p>The Sessions High count for Web Applications is not accurate, and only displays counts from the time the server was started.</p> <p><b>Note:</b> The Open Sessions count, which previously had this problem, was recently fixed and is accurate.</p>
CR172752	<p>During Web application undeployment, a servlet will throw NPEs (NullPointerExceptions). This is a Sun servlet specification limitation. To avoid this, the application code should call <code>"super.init(ServletConfig)"</code> in the servlet <code>init()</code> call.</p> <p><b>Note:</b> This a known limitation.</p>

## Transactions Known Issues

Change Request Number	Description
CR130269	<p>By default, database connections that use an XA driver have <code>autoCommit</code> set to <code>false</code>. If you use a database connection that is based on an XA driver for a read-only operation, the connection implicitly creates a local transaction. When the connection is returned to the connection pool, the local transaction still exists on the connection. The next time the connection is used, WebLogic Server internally calls <code>XAResource.start()</code>, which fails because there is already a transaction on the connection.</p> <p><b>Workaround:</b> Set <code>autoCommit</code> to <code>true</code> or call <code>connection.commit()</code> before returning the connection to the connection pool.</p>

## WebLogic JDBC Type 4 Drivers Known Issues

Change Request Number	Description
n/a	<p data-bbox="391 453 505 482"><b>DB2 Driver</b></p> <ul data-bbox="391 493 1077 921" style="list-style-type: none"> <li data-bbox="391 493 1077 609">• Because of lack of support in the DRDA listener for all platforms except DB2 UDB v8.1, ResultSetMetaData describes BIGINT columns as DECIMAL, and stored procedures accepting BIGINT as an input parameter fail on execution with a conversion error.</li> <li data-bbox="391 618 1077 734">• Scroll-sensitive result sets are not supported. Requests for scroll-sensitive result sets are downgraded to scroll-insensitive result sets when possible. When this happens, a warning is generated indicating that result set scrollability has been downgraded.</li> <li data-bbox="391 743 1077 921">• The WebLogic Type 4 JDBC DB2 driver must be able to determine the data type of the column or stored procedure argument to implicitly convert the parameter value. Not all DB2 database versions support getting parameter metadata for prepared statements. Implicit conversions are not supported on database versions that do not provide parameter metadata for prepared statements.</li> </ul>
n/a	<p data-bbox="391 944 723 973"><b>DB2 Driver on Windows and UNIX</b></p> <ul data-bbox="391 982 1077 1107" style="list-style-type: none"> <li data-bbox="391 982 1077 1039">• Except for DB2 UDB 8.1, CLOB data types are limited to 32 K characters because of DRDA listener limitations.</li> <li data-bbox="391 1048 1077 1107">• Except for DB2 UDB 8.1, BLOB data types are not supported because of DRDA listener limitations.</li> </ul>
n/a	<p data-bbox="391 1130 723 1159"><b>DB2 Driver on iSeries and AS/400</b></p> <ul data-bbox="391 1168 1077 1229" style="list-style-type: none"> <li data-bbox="391 1168 1077 1229">• Except for DB2 iSeries V5R2, BLOB and CLOB data types are not supported.</li> </ul>
n/a	<p data-bbox="391 1251 602 1281"><b>MS SQL Server Driver</b></p> <ul data-bbox="391 1289 1077 1350" style="list-style-type: none"> <li data-bbox="391 1289 1077 1350">• To ensure correct handling of character parameters, install Service Pack 2, or higher, for SQL Server 7 installations.</li> </ul>
CR111002	<p data-bbox="391 1373 1077 1517">When using a configuration that has more than one connection pool that uses the WebLogic Type 4 JDBC driver for MS SQL Server, it is possible that WebLogic Server may hang during transaction recovery. This is due to a problem in Microsoft Distributed Transaction Coordinator (DTC) and SQL Server.</p> <p data-bbox="391 1527 1013 1558">Until a hotfix is publicly available from Microsoft, call BEA Support.</p>

Change Request Number	Description
n/a	<p data-bbox="455 388 588 409"><b>Oracle Driver</b></p> <ul data-bbox="455 427 1139 933" style="list-style-type: none"> <li data-bbox="455 427 1139 508">• If you want to take advantage of JDBC distributed transactions through JTA with the Oracle driver, you must install Oracle version 8.1.7 or higher, and install the Oracle JAVA_XA package on the database server.</li> <li data-bbox="455 526 1139 690">• Because JDBC does not support a cursor data type, the Oracle driver returns refcursor output parameters to the application as result sets. Your application should not include a parameter marker or should not declare the refcursor output parameter. The driver transparently converts the refcursor to a result set, which can be retrieved using <code>getResultSet</code> or <code>getMoreResults</code>.</li> <li data-bbox="455 708 1139 933">• By default, values for <code>TIMESTAMP WITH TIME ZONE</code> columns cannot be retrieved using the <code>ResultSet.getTimeStamp</code> method because the time zone information would be lost. The driver returns <code>NULL</code> when the <code>getTimeStamp</code> method is called on a <code>TIMESTAMP WITH TIME ZONE</code> column. It generates a <code>SQLException</code> in that case. For details about using the <code>TIMESTAMP WITH TIME ZONE</code> data type with the Oracle driver, see "<a href="#">TIMESTAMP WITH TIME ZONE Data Type</a>" in <i>WebLogic Type 4 JDBC Drivers</i>.</li> </ul>
CR120330	<p data-bbox="455 965 588 986"><b>Sybase Driver</b></p> <p data-bbox="455 1003 1139 1116">The <code>getDatabaseProductName</code> method returns "SQL Server," which is the string returned internally by the Sybase database server. This value may not be the same return as seen with other JDBC drivers, including the Sybase JConnect JDBC drivers.</p> <p data-bbox="455 1133 1139 1187"><b>Workaround:</b> If you need the return string to include "Sybase," use the <code>getDriverName</code> method.</p>

Change Request Number	Description
CR123475	<p data-bbox="391 387 561 409"><b>Sybase XA Driver</b></p> <p data-bbox="391 427 1072 510">When using the Sybase XA driver connecting to a Sybase 12.5.0.3 database and calling <code>getMetaData()</code> after some DDL operations, the method erroneously returns <code>null</code>.</p> <p data-bbox="391 527 518 550"><b>Workaround:</b></p> <p data-bbox="391 567 1049 619">In the connection properties, set <code>useAlternateMetaData=true</code>. For example:</p> <p data-bbox="391 637 881 659">In a connection pool, the configuration may look like:</p> <pre data-bbox="391 677 1072 956"> &lt;JDBCConnectionPool DriverName="weblogic.jdbcx.sybase.SybaseDataSource" Name="MyJDBCConnectionPool" Password="{3DES}x1KgkDjghuQ=" Properties="user=scott;url=jdbc:bea:sybase://sybserver:4100; <b>useAlternateMetaData=true</b>;PortNumber=4100;ServerName=sybserver;DatabaseName=wl" Targets="mycluster" URL="jdbc:bea:sybase://sybserver:4100"/&gt; </pre> <p data-bbox="391 973 830 996">In a non-pooled connection, code may look like:</p> <pre data-bbox="391 1013 1072 1263"> SybaseDataSource ds = new SybaseDataSource(); ds.setUser("USER"); ds.setPassword("PASSWORD"); ds.setServerName("SERVER"); ds.setPortNumber(5000); <b>ds.setExtendedOptions("useAlternateMetaData=true");</b> xaConn = ds.getXAConnection(); conn = xaConn.getConnection(); </pre>

Change Request Number	Description
n/a	<p data-bbox="455 390 561 413"><b>All Drivers</b></p> <ul style="list-style-type: none"> <li data-bbox="455 427 1139 626">• The WebLogic Type 4 JDBC drivers support retrieval of output parameters from a stored procedure before all result sets and/or update counts have been completely processed. When <code>CallableStatement.getXXX</code> is called, result sets and update counts that have not yet been processed by the application are discarded to make the output parameter data available. Warnings are generated when results are discarded.</li> <li data-bbox="455 640 1139 812">• The WebLogic Type 4 JDBC drivers allow <code>PreparedStatement.setXXX</code> and <code>ResultSet.getXXX</code> methods on BLOB/CLOB data types, in addition to what is described in the JDBC specification. The supported conversions are generally the same as those for LONGVARBINARY/LONGVARCHAR, except where limited by database support.</li> <li data-bbox="455 826 1139 1055">• Calling <code>CallableStatement.registerOutputParameter(parameterIndex, sqlType)</code> with <code>sqlType</code> <code>Types.NUMERIC</code> or <code>Types.DECIMAL</code> sets the scale of the output parameter to zero. According to the JDBC specification, calling <code>CallableStatement.registerOutputParameter(parameterIndex, sqlType, scale)</code> is the recommended method for registering NUMERIC or DECIMAL output parameters.</li> <li data-bbox="455 1069 1139 1177">• When attempting to create an updatable, scroll-sensitive result set for a query that contains an expression as one of the columns, the driver cannot satisfy the scroll-sensitive request. The driver downgrades the type of the result returned to scroll-insensitive.</li> <li data-bbox="455 1190 1139 1454">• The preferred method for executing a stored procedure that generates result sets and update counts is using <code>CallableStatement.execute()</code>. If multiple results are generated using <code>executeUpdate</code>, the first update count is returned. Any result sets prior to the first update count are discarded. If multiple results are generated using <code>executeQuery</code>, the first result set is returned. Any update counts prior to the first result set are discarded. Warnings are generated when result sets or update counts are discarded.</li> <li data-bbox="455 1468 1139 1605">• The result set methods <code>getTimestamp</code>, <code>getDate</code>, and <code>getTime</code> return references to mutable objects. If the object reference returned from one of these methods is modified, then re-fetching the column using the same method will return the modified value. The value is only modified in memory; the database value is not modified.</li> </ul>

Change Request Number	Description
CR124744	<p><b>All Drivers</b></p> <p>When calling <code>Statement.setQueryTimeout()</code>, SQL statements take twice as long as specified to time out.</p> <p><b>Workaround:</b> Set the query timeout to half as long as desired.</p> <p>This problem was resolved in 8.1 SP3.</p>
CR178619	<p><b>XA Driver for DB2 and Microsoft SQL Server</b></p> <p>Drivers return the error code of RMERR (-3) instead of NOTA (-4) which causes transactions to be retried until <code>AbandonTimeoutSeconds</code> instead of the transaction being resolved on recovery.</p>

## Web Services Known Issues

Change Request Number	Description
CR127610	<p>When invoking a WebLogic Web Service operation using the stubs generated by the <code>clientgenAnt</code> task, the client application might sometimes throw a <code>ClassCastException</code> error if the method that implements the operation uses a global element of a global complex type. The problem is that the <code>clientgenAnt</code> task, when generating the stubs, uses the wrong parameter class due to the data type mapping information in the provided <code>types.xml</code> file being overwritten by JAX-RPC default.</p>
CR128123	<p>A WebLogic Server 8.1 clientgen-generated Web Service that uses SSL transport is not supported on a target client platform based on a JVM version below JDK 1.4.1.</p>

Change Request Number	Description
CR131690	<p>When sending a SOAP request that contains encrypted multi-byte strings to the Web Service, the Web Service client cannot specify <code>Shift_JIS</code> or <code>EUC-JP</code> into the message's encoding <code>weblogic.webservice.i18n.charset</code>. If those encodings are specified, the Web Service returns a <code>SOAPFaultException</code>. When using <code>UTF-8</code>, this issue does not occur.</p> <p><b>Workaround:</b> Do not set <code>Shift_JIS</code> and <code>EUC-JP</code> in <code>weblogic.webservice.i18n.charset</code> when sending encrypted multi-byte strings. Set <code>UTF-8</code> instead.</p>
CR175036	<p>If a WSDL has elements with the same name but with a difference in the case (upper or lower) of one or more letters, running <code>clientgen</code> on the WSDL fails with an error such as:</p> <pre data-bbox="455 786 1137 1060">C:\mydir\resources&gt;java weblogic.webservice.clientgen -packageName cargoPackage -wsdl WSDLName.wsdl -clientJar case.jar C:\DOCUME~1\anurag\LOCALS~1\Temp\case.jar-1007802 369\org\openuri\www\FooCodec.java:14: class FOOCode c is public, should be declared in a file named FOOCodec.java public final class FOOCodec</pre> <p>This is a system limitation of Windows 32-bit operating systems. You cannot generate <code>FOO.java</code> and <code>Foo.java</code> in the same Windows (32-bit) directory because the operating system is case-insensitive. The second file will always overwrite the first one.</p>

## Known Issues in WebLogic Server 8.1 SP1

The following sections describe known issues in WebLogic Server 8.1 SP1:

- [“Administration Console Known Issues” on page 2-47](#)
- [“Cluster Known Issues” on page 2-47](#)
- [“Connector Known Issues” on page 2-48](#)
- [“EJB Known Issues” on page 2-49](#)
- [“Examples Known Issues” on page 2-50](#)
- [“JVM Known Issues” on page 2-51](#)
- [“Miscellaneous Known Issues” on page 2-52](#)
- [“Node Manager Known Issues” on page 2-52](#)
- [“Plug-Ins Known Issues” on page 2-53](#)
- [“Security Known Issues” on page 2-53](#)
- [“Servlets & JSPs Known Issues” on page 2-55](#)
- [“System Administration Known Issues” on page 2-55](#)
- [“WebLogic JDBC Type 4 Driver for MS SQL Server Known Issues” on page 2-56](#)
- [“WebLogic Workshop Known Issues” on page 2-59](#)
- [“Web Services Known Issues” on page 2-59](#)
- [“WLEC Known Issues” on page 2-61](#)
- [“WTC Known Issues” on page 2-61](#)

## Administration Console Known Issues

Change Request Number	Description
CR111676	<p>The navigation tree in the left pane of the WebLogic Server Administration Console will not display if it is running directly on a system with the following configurations:</p> <ul style="list-style-type: none"> <li>• Windows 64-bit platform with Internet Explorer 6</li> <li>• Linux 64-bit platform with Mozilla 1.0.1</li> </ul> <p><b>Workarounds:</b></p> <ul style="list-style-type: none"> <li>• Obtain the Java plug-in from the operating system vendor's Web site. If the plug-in is not available, run the Administration Console on a system with a configuration that supports the Java plug-in.</li> <li>• Run the Administration Console on a system with a different configuration.</li> </ul>

## Cluster Known Issues

Change Request Number	Description
CR105861	<p>When a Web application and EJB are deployed in two different clusters and an EJB stub is placed into the HTTP session, a <code>java.lang.ClassCastException</code> is thrown.</p> <p><b>Workaround:</b></p> <p>Rather than putting the stub into the HTTP session, you should put the <code>Handle</code> into the HTTP session and also ensure that any custom application classes are in the system classpath. Thereafter, you can obtain the <code>EJBObject</code> from this <code>Handle</code>.</p>

## Connector Known Issues

---

<b>Change Request Number</b>	<b>Description</b>
CR091818	Some customers would like to disable the proxy test that is automatically invoked the first time a connection to the back-end system is made.  This problem was resolved in 8.1 SP2.

---

## EJB Known Issues

Change Request Number	Description
CR105964	<p>When XML Spy is installed, DTD file types are automatically associated with XML Spy. Therefore, when customers open the DTD in a browser, XML Spy attempts to validate the DTD, and produces the following error:</p> <pre data-bbox="454 562 1139 869"> === Error message === The XML page cannot be displayed Cannot view XML input using style sheet. Please correct the error and then click the Refresh button, or try again later. ----- ----- Cannot have a DTD declaration outside of a DTD. Error processing resource 'http://www.bea.com/servers/wls810/dtd/weblogic-r dbms20-persistence-810.dtd'. Line 13, Position 11 &lt;!ELEMENT weblogic-rdbms-jar ( -----^ </pre> <p><b>Workarounds:</b></p> <ul data-bbox="454 925 1139 1055" style="list-style-type: none"> <li>• View the file in XML Spy (using the Tools → Edit option). When viewing the file in XML Spy, it can also be saved on disk using Save As option.</li> <li>• Delete the DTD file type association (using the File Explorer → Tools → Folder Options → File Types option).</li> </ul>
CR106907	<p>Some XA JDBC drivers do not support local transaction operations, which can cause an error similar to the following when optimistic concurrency is used with such a driver:</p> <pre data-bbox="454 1180 1079 1230"> SQL operations are not allowed with no global transaction by default for XA drivers. </pre> <p>In other words, the error will occur when <code>SupportsLocalTransactions="true"</code> is specified for the <code>JDBCConnectionPool</code>.</p> <p>This problem occurs because optimistic concurrency suspends a global transaction and does reads in a local transaction when the database is not Oracle. (When using Oracle, you can avoid this problem by explicitly setting <code>&lt;database-type&gt;Oracle&lt;/database-type&gt;</code> in your CMP deployment descriptor.)</p> <p><b>Workaround:</b> Use the <code>"RollbackLocalTxUponConnClose"</code> on the <code>JDBCConnection</code>.</p>

## Examples Known Issues

Change Request Number	Description
CR072190	<p>A known Tuxedo issue (CR050182) prevents Tuxedo clients and servers from compiling on HP platforms for the following example packages:</p> <ul style="list-style-type: none"><li>• <code>iiop.ejb.stateless.tuxclient</code></li><li>• <code>iiop.ejb.stateless.txtuxclient</code></li><li>• <code>iiop.ejb.entity.tuxclient</code></li><li>• <code>iiop.ejb.statelss.tux</code></li></ul> <p><b>Workaround:</b> Use a platform other than HP.</p>
CR108402	<p>In Step 3 of the instructions in the <code>package-summary.html</code> file for the <code>examples.jsp.tagext.form_validation</code> example, the directory to which you must change is incorrect. The correct directory is <code>SAMPLES_HOME\server\examples\src\examples\jsp\tagext\form_validation</code>.</p> <p>This problem was resolved in 8.1 SP2.</p>
CR121588	<p>The MedRec server fails to start on WebLogic Server 8.1 SP1 on Solaris 8 and Windows 2003 SP3 while attempting to connect to <code>jakarta.apache.org:80</code>. This issue occurs because of a bug in Struts, documented at <a href="http://www.mail-archive.com/struts-user@jakarta.apache.org/msg60534.html">http://www.mail-archive.com/struts-user@jakarta.apache.org/msg60534.html</a>.</p> <p><b>Workaround:</b> A connection to <code>apache.org</code> should be available during server startup.</p>

## JVM Known Issues

---

Change Request Number	Description
CR099962	<p data-bbox="455 461 1137 545">When running on the IA64 platform with the JRockit JVM, non-trivial Web applications (that is, those with complex JSPs) experience performance problems when first loaded.</p> <p data-bbox="455 562 1137 736">This problem occurs because JRockit is a compile-only JVM, and consequently, it has problems at first with very large methods such as those being generated from a complex JSP. The first time such a JSP is hit, the JVM will generate the native code for the method, and this can take ten of seconds. Note that reloads will not be slow. This problem typically occurs the first time a server is booted.</p> <p data-bbox="455 753 1137 833"><b>Workaround:</b> Non-trivial Web applications either should not be run on JRockit on the IA64 platform, or they will just perform slower than usual (only on the first load).</p>

---

## Miscellaneous Known Issues

Change Request Number	Description
CR104392	<p data-bbox="391 453 1077 574">Under stress conditions on Linux 64, sockets can be left in a <code>CLOSE_WAIT</code> state even after they are explicitly closed. This could possibly be due to socket buffer overflow, which may cause packet loss during the final TCP handshake to close the connection.</p> <p data-bbox="391 583 518 612"><b>Workaround:</b></p> <p data-bbox="391 621 1077 685">The workaround is to increase the amount of memory the system allocates for each TCP socket connection as follows:</p> <ul data-bbox="391 694 680 982" style="list-style-type: none"> <li data-bbox="391 694 646 723">• <code>tcp_rmem</code>—128388607</li> <li data-bbox="391 732 646 762">• <code>tcp_wmem</code>—128388607</li> <li data-bbox="391 770 633 800">• <code>rmem_max</code>—4194303</li> <li data-bbox="391 808 633 838">• <code>wmem_max</code>—4194303</li> <li data-bbox="391 847 680 876">• <code>tcp_keepalive_time</code>—1800</li> <li data-bbox="391 885 653 914">• <code>tcp_window_scaling</code>—0</li> <li data-bbox="391 923 545 953">• <code>tcp_sack</code>—0</li> <li data-bbox="391 961 612 991">• <code>tcp_timestamps</code>—0</li> </ul>

## Node Manager Known Issues

Change Request Number	Description
CR109093	<p data-bbox="391 1218 1077 1312">When restarted, Node Manager requires the generated <code>NodeManagerLogs/NodeManagerInternal/NodeManagerProperties</code> file. It uses the seed in this file to decrypt the password.</p> <p data-bbox="391 1321 1077 1381">Please do not delete this file. Users could accidentally delete this file since it is under the <code>logs</code> directory.</p>

## Plug-Ins Known Issues

Change Request Number	Description
CR102290	This version of WebLogic Server does not support Apache plug-ins for the Windows 64 platform because the Apache binary distribution is not yet available.

## Security Known Issues

For information about BEA security advisories, refer to the BEA Advisories & Notifications page on the dev2dev Web site. On this page, you can download security-related patches and register to receive notifications of newly available security advisories. You can access the Advisories & Notifications page with the following URL: <http://dev2dev.bea.com/advisories>.

BEA has established an e-mail address ([security-report@bea.com](mailto:security-report@bea.com)) to which you can send reports of any possible security issues in BEA products.

The following table summarizes security-related known issues for the 8.1 SP1 release.

Change Request Number	Description
CR100789	Using nCipher with JRockit for the IIOP thin client is not working properly. <b>Workaround:</b> Do not use nCipher in this environment.
CR103330	The <code>createPolicy</code> method inherited from the <code>PolicyEditorMBean</code> optional SSPI is not case sensitive. Therefore, two resources with <code>resourceIDs</code> differing only by case can not be created within a WebLogic Authorization provider.

---

<b>Change Request Number</b>	<b>Description</b>
CR110389	The <code>weblogic.Admin</code> and <code>weblogic.Deployer</code> methods can be used with a username and password. If you specify a username on the command line but do not specify a password, they will prompt for one. By default, the password is not echoed to the screen as you type it. If you run the admin command without having set the client's path to include <code>...\weblogic81\server\bin</code> in your PATH, then the native libraries that perform the non-echoing input cannot be found and <code>weblogic.Admin</code> and <code>weblogic.Deployer</code> will echo the password as you type it. By default, all WebLogic Server start scripts include this directory in the path so this is only an issue if you are using custom scripts or no scripts at all.
CR113469	Due to an unintentional change in the source code, security providers compiled for releases prior to WebLogic Server 8.1 will need to be recompiled for 8.1 to run.

---

## Servlets & JSPs Known Issues

Change Request Number	Description
CR106460	<p>Previous versions of WebLogic Server resolved URIs that contained extra spaces. WebLogic Server 8.1 does not resolve extra spaces, and a URI request that contains extra spaces will result in a 404. For example, previous versions of WebLogic Server resolved <code>http://server:port/mywebapp/foo%20%20</code> to the resource <code>foo</code> in the Web application <code>mywebapp</code>, but beginning with 8.1 it no longer does.</p>
CR111102	<p>The <code>save-sessions-enabled</code> option to maintain <code>HttpSession</code> state across redeployments does not work if the session persistence type is set to <code>Replicated</code>.</p>

## System Administration Known Issues

Change Request Number	Description
CR110835	<p>Configuration objects created during server startup are not always persisted.</p> <p>If you use a startup class to create configuration objects (such as JDBC Connection Pools or any other resource whose configuration is represented by MBeans), WebLogic Server does not save the configuration data to <code>config.xml</code> unless you modify the configuration after the server startup cycle.</p> <p><b>Workaround:</b> Instead of using startup classes to create configuration objects, use <code>weblogic.Admin</code>, <code>wlconfig</code>, or the Administration Console after the server has completed its startup cycle.</p>

## WebLogic JDBC Type 4 Driver for MS SQL Server Known Issues

Change Request Number	Description
n/a	<p>Before you can use distributed transactions with the WebLogic JDBC Type 4 Driver for MS SQL Server, you must install JDBC XA stored procedures in the database. See <a href="#">“Installing Stored Procedures for JTA”</a> in <i>BEA Weblogic Type 4 JDBC Drivers</i>.</p>
n/a	<p>Microsoft SQL Server 7 does not allow resource sharing because it cannot release the connection to a transaction until it commits or rolls back. For example, the following code will fail:</p> <pre>xaResource.start(xid1, TMNOFLAGS) ... xaResource.end(xid1, TMSUCCESS) xaResource.start(xid2, TMNOFLAGS) ---&gt; fail</pre> <p><b>Workaround:</b> Set <code>keepXAConnTillTxComplete=true</code> on the JDBC connection pool.</p>
n/a	<p>When a global transaction is suspended or ended on a connection to a Microsoft SQL Server database, the database does not disassociate the transaction with subsequent database operations. If the global transaction is rolled back, all other database operations that were done since the transaction started will be rolled back with the transaction. For example, in the following code, the <code>insert</code> operation on <code>Table2</code> is rolled back even though it is outside the scope of the transaction:</p> <pre>xaResource.start(xid1, TMNOFLAGS) stmt.executeUpdate("insert into table1 values (1)"); xaResource.end(xid1, TMSUCCESS)  stmt.executeUpdate("insert into table2 values (2)");  xaResource.prepare(xid1); xaResource.rollback(xid1);</pre>

Change Request Number	Description
n/a	When requesting an updatable <code>ResultSet</code> (by specifying <code>ResultSet.CONCUR_UPDATEABLE</code> ) and a server-side cursor is created, the updatable <code>ResultSet</code> has pessimistic-locking semantics. If a server-side cursor is not created, the updatable <code>ResultSet</code> has optimistic-locking semantics. A server-side cursor may be created if <code>selectMethod</code> is set to <code>cursor</code> or if a scrollable <code>ResultSet</code> is requested. Ultimately, the transaction isolation level should be used to manage concurrency issues.
n/a	The WebLogic JDBC Type 4 Driver for MS SQL Server supports retrieval of output parameters from a stored procedure before all <code>ResultSets</code> or update counts have been completely processed. When <code>CallableStatement.getXXX</code> is called, <code>ResultSets</code> and update counts yet to be processed by the application are discarded to make the output parameter data available. Warnings are generated when results are discarded.
n/a	When attempting to create an updatable, scroll-sensitive <code>ResultSet</code> for a query that contains an expression as one of the columns, the driver cannot satisfy the scroll-sensitive request. The driver downgrades the type of the result returned to scroll-insensitive.
n/a	The preferred method for executing a stored procedure that generates <code>ResultSets</code> and update counts is using <code>CallableStatement.execute()</code> . If multiple results are generated using <code>executeUpdate</code> or <code>executeQuery</code> , the first update count or <code>ResultSet</code> , respectively, in the multiple result is returned. All other results are discarded. Warnings are generated when results are discarded.
n/a	The WebLogic JDBC Type 4 Driver for MS SQL Server allows <code>PreparedStatement.setXXX</code> methods and <code>ResultSet.getXXX</code> methods on BLOB/CLOB types above what is described in the JDBC specification. The supported conversions are generally the same as those for LONGVARBINARY/LONGVARCHAR, except where limited by database support.
n/a	If you are using Microsoft SQL Server 7, you should install Service Pack 2 or higher on the database server to ensure correct handling of character parameters.

<b>Change Request Number</b>	<b>Description</b>
n/a	Because of the way CHAR, VARCHAR, and LONGVARCHAR data types are handled internally by the driver, parameters of these data types exceeding 4000 characters in length cannot be compared or sorted, except when using the IS NULL or LIKE operators.
n/a	The ResultSet methods getTimestamp, getDate, and getTime return References to mutable objects. If the object reference returned from one of these methods is modified, then refetching the column using the same method will return the modified value. The value is only modified in memory; the database value is not modified.

## WebLogic Workshop Known Issues

Change Request Number	Description
n/a	Although the WebLogic Workshop software is distributed with WebLogic Server software, neither the WebLogic Workshop IDE nor the WebLogic Workshop runtime framework is supported on IA64 systems on any operating system.

## Web Services Known Issues

Change Request Number	Description
n/a	By default, the result of running <code>clientgen</code> in WebLogic Server 8.1 is <code>.class</code> files rather than <code>.java</code> files. This is a change in behavior from running <code>clientgen</code> in WebLogic Server 7.0. To make <code>clientgen</code> generate <code>.java</code> files in WebLogic Server 8.1, use the <code>keepgenerated="true"</code> attribute in <code>clientgen</code> . Documentation for this attribute is available at <a href="#">“clientgen”</a> in <i>Programming WebLogic Web Services</i> .
CR100413	When using Web Services and two-way SSL, identity assertion is required whether or not the Web Service resource is protected. It should be possible to configure WebLogic Server to establish link-level trust without requiring identity assertion for Web Services that do not require authentication.  <b>Workaround:</b> Clients should only use certificates when required to access the service (to avoid unnecessary overhead) and care should be taken that all valid client certificates map to valid WebLogic Server users.
CR105388	When XML encodings are sent as a SOAP attachment, the message is not sent correctly.

---

<b>Change Request Number</b>	<b>Description</b>
CR107934	<p>Even if an <code>encryptKeyName</code> is not provided as part of the <code>&lt;servicegen&gt;</code> task, an <code>EncryptBody</code> attribute with the value equal to "true" is added to the generated <code>web-services.xml</code> file. This indicates that the SOAP message should be encrypted.</p> <p><b>Workaround:</b> Edit the generated <code>web-services.xml</code> deployment descriptor for the Web Service to set the <code>EncryptBody="false"</code>.</p> <p>This problem was resolved in 8.1 SP3.</p>
CR108646	<p>The <code>&lt;servicegen&gt;</code> task does not support the "mergewithexistingws" attribute as described in the documentation.</p> <p><b>Workaround:</b> Specify different <code>destEar</code> and manually merge the two <code>web-services.xml</code> files.</p>

---

## WLEC Known Issues

Change Request Number	Description
CR106393	<p>In 8.1, the classes packaged in <code>wleorb.jar</code> will conflict with those in the JDK (<code>rt.jar</code>). Customers who use WLEC (which is deprecated) will experience an exception when WLEC connections are attempted.</p> <p><b>Workaround:</b></p> <p>Prepend <code>wleorb.jar</code> to the JVM bootclasspath using <code>-Xbootclasspath/p:\$WL_HOME/lib/wleorb.jar</code> to get WLEC working properly. Note, however, that doing this will cause problems for customers also wanting to use IOP. In this case, customers should use WebLogic Tuxedo Connector instead.</p>

## WTC Known Issues

Change Request Number	Description
CR106305	The <code>examples/wtc/atmi/simpview</code> example fails to run.
CR111102	<p><b>Workaround:</b></p> <p>In the WebLogic Server Configuration section, add the <code>infoenc</code> VIEW table to the server classpath before completing Step 1.</p> <p>One method is to update the server class path by modifying the <code>startExamplesServer</code> script located at <code>SAMPLES_HOME\domains\examples.NT/2000</code> users modify the <code>startExamplesServer.cmd</code> script. Unix users modify the <code>startExamplesServer.sh</code> script.</p> <p><b>Example:</b> <code>set CLASSPATH=%WEBLOGIC_CLASSPATH%;%CLASSPATH%;%SAMPLES_HOME%\server\examples\src\examples\wtc\atmi\simpview\infoenc</code></p>
CR109849	<p>WTC 8.1 does not handle VIEW buffers from Tuxedo 6.5 services. This problem occurs because of a logic error in one of the internal WTC routines that deal with Tuxedo 6.5 compatibility.</p> <p>This problem was resolved in 8.1 SP2.</p>

## Known Issues in WebLogic Server 8.1

The following sections describe known issues in WebLogic Server 8.1:

- [“Administration Console Known Issues” on page 2-64](#)
- [“Classloader Known Issues” on page 2-66](#)
- [“Configuration Wizard Known Issues” on page 2-67](#)
- [“CORE Known Issues” on page 2-68](#)
- [“Deployment Known Issues” on page 2-69](#)
- [“EJB Known Issues” on page 2-69](#)
- [“Examples Known Issues” on page 2-72](#)
- [“Installation Known Issues” on page 2-74](#)
- [“Interoperability Known Issues” on page 2-74](#)
- [“J2EE Known Issues” on page 2-76](#)
- [“JDBC Known Issues” on page 2-76](#)
- [“JMS Known Issues” on page 2-79](#)
- [“JVM Known Issues” on page 2-80](#)
- [“Oracle Thin Driver Known Issues” on page 2-81](#)
- [“RMI/RMI-IIOP Known Issues” on page 2-83](#)
- [“Security Known Issues” on page 2-85](#)
- [“Servlets & JSPs Known Issues” on page 2-89](#)
- [“Servlets & JSPs Known Issues” on page 2-89](#)
- [“Tools Known Issues” on page 2-91](#)
- [“WebLogic jDriver for Oracle Known Issues” on page 2-92](#)
- [“WebLogic Tuxedo Connector Known Issues” on page 2-93](#)
- [“WebLogic Workshop Known Issues” on page 2-95](#)

- [“Web Services Known Issues” on page 2-95](#)
- [“XML Known Issues” on page 2-96](#)

## Administration Console Known Issues

Change Request Number	Description
CR079374	If you add an application module such as a Web Application to a Enterprise Application (EAR) already deployed in your domain, you must delete the Enterprise Application from the domain configuration and then add it again in order for the module to become active.
CR080476	You designate a default Web Application with the Web Application's deployment descriptor. You no longer use the Administration Console. See <a href="#">"Deploying a Default Web Application"</a> in the <i>Administration Console Online Help</i> .
CR088462	There are problems setting targets for EJB deployments using Internet Explorer version 5.0. Internet Explorer version 5.0 is not supported for the Administration Console. Please use version 5.5 or 6.0.
CR091141	Netscape version 4.79 does not work correctly on Linux AS 2.1. Netscape 4.79 on Linux AS 2.1 is not supported for use with the Administration Console. <b>Workaround:</b> Use the Mozilla 1.0 browser, which is supported for Linux AS2.1.
CR099866	Administration Console extensions (see <a href="#">"Extending the Administration Console"</a> ) do not function correctly if you use enable a domain-wide administration port (see <a href="#">"Enabling the Domain-Wide Administration Port"</a> in the <i>Administration Console Online Help</i> ).
CR100159	The instructions for unlocking a user account in the Security section of the Administration Console online help are incorrect. <b>Workaround:</b> To unlock a user account: <ol style="list-style-type: none"> <li>1. Expand the Servers--&gt;Monitoring tab.</li> <li>2. Click the Security tab.</li> <li>3. In the Unlock User attribute, enter the user name for a user of this server who has been locked out.</li> <li>4. Click Apply.</li> <li>5. If the unlock was successful, a confirmation message appears at the top of the Monitoring--&gt;Security page.</li> </ol>

---

Change Request Number	Description
CR100460	<p>When you use the Administration Console to clone a server that is a target of an existing JMS distributed queue or topic, the following exception is displayed in the console:</p> <pre data-bbox="434 487 1094 539">javax.management.InvalidAttributeValueException: Illegal target</pre> <p>Although the newly cloned server is created, it does not appear in the Navigation Tree, but does appear in the summary server view.</p> <p><b>Workaround:</b> To successfully clone a server that is the target of an existing queue or topic, do not use the clone function. Instead, create a new server and manually configure the server.</p>
CR101317	<p>When using the Administration Console with Internet Explorer 5.5, you may see the following JavaScript error message: "Object does not support this property or method". Upgrading to Internet Explorer 6.0 corrects this problem. This error appears only occasionally on some computers where unspecified Internet Explorer patches have been applied.</p>

---

## ClassLoader Known Issues

---

Change Request Number	Description
CR094307	<p data-bbox="391 461 1072 690">A <code>ClassCastException</code> might occur if you change the classloader hierarchy. For example, consider an application that has a Web application and an EJB. The Web application has a <code>ejb-local-ref</code> to the EJB. If the default classloader structure is used, the Web Application is able to access the EJB using the <code>ejb-ref</code>. However, if the classloader structure is changed so that the Web application classloader and the EJB classloader become peers, a <code>ClassCastException</code> is thrown, since the EJB class is no longer loaded from the common parent classloader.</p> <p data-bbox="391 708 1072 821"><b>Workaround:</b> The two Web application and EJB must share classloaders or the called module must have a classloader that is an ancestor of the calling module. The server fails the deployment if you provide a local reference to a module in a classloader that is a peer or offspring of the calling module.</p>

---

## Configuration Wizard Known Issues

Change Request Number	Description
n/a	<p>When configuring a JDBC connection pool, you can only test the connection configuration parameters if you are using a JDBC driver installed with WebLogic Server.</p> <p><b>Workaround:</b></p> <p>To test connection configuration parameters for other drivers, you can change the script that launches the Configuration Wizard to include other drivers in the CLASSPATH. Follow these steps:</p> <ol style="list-style-type: none"> <li>1. Open <code>WL_HOME\common\bin\config.cmd</code> in a text editor (where <code>WL_HOME</code> is the folder where you installed WebLogic Server, typically <code>C:\bea\weblogic81</code>).</li> <li>2. Find the following lines: <pre>if "%1" NEQ "" goto java "%JAVA_HOME%\bin\javaw" %JAVA_VM% -jar "c:\bea\weblogic81\common\lib\config.jar" goto end :java "%JAVA_HOME%\bin\java" %JAVA_VM% -jar "c:\bea\weblogic81\common\lib\config.jar" %*</pre> </li> <li>3. Replace those lines with the following single line: <pre>"%JAVA_HOME%\bin\java" %JAVA_VM% -cp .;config.jar;3rdparty.jar;comdev.jar;wizard.jar;%WL_HOME%\server\lib\weblogic.jar;%WL_HOME%\common\eval\pointbase\lib\pbserver44.jar;%WL_HOME%\common\eval\pointbase\lib\pbtools44.jar;%WL_HOME%\common\eval\pointbase\lib\pbclient44.jar com.bea.plateng.wizard.WizardController</pre> </li> <li>4. Add the path to additional Type 4 JDBC drivers to the end of the <code>-cp</code> string (after <code>pbclient44.jar</code>).</li> </ol> <p><b>Note:</b> This change only affects the Configuration Wizard when you start it using the <code>config.cmd</code> file. It does not take affect when you start the Configuration Wizard from the Start menu.</p> <p>These instructions are for a Windows platform. You can adapt them for a UNIX platform.</p>

## CORE Known Issues

---

<b>Change Request Number</b>	<b>Description</b>
CR101946	If you use the Java socket reader implementation, extended use of JMS can lead to an Out of Memory error.  <b>Workaround:</b> Use the native WebLogic Server performance pack. See <a href="#">WebLogic Server Performance and Tuning</a> .
CR102874	By default, network channel tunneling is on, which may pose a security risk if you have explicitly configured network channels for a WebLogic Server instance.  This problem is resolved in 8.1 SP1, where network channel tunneling is disabled by default.

---

## Deployment Known Issues

Change Request Number	Description
CR096834	<p>If a member of a WebLogic Server cluster shuts down or becomes unavailable when you are deploying a large application to the cluster, the deployment task can appear to hang. This occurs because the two-phase deployment protocol is waiting for a response from the server that has shut down. The task will appear to hang until the default timeout length of 1 hour is reached.</p> <p><b>Workaround:</b> If you anticipate changes to a cluster during a deployment, use the <code>-timeout</code> option to <code>weblogic.Deployer</code> to specify a shorter timeout period.</p>
CR100540	<p>WebLogic Server uses inconsistent module names when you deploy archive files and exploded archive directories. When you deploy an archive file, its module names are not registered using the file extension. For example, consider the following case where you deploy an Enterprise Application, <code>myapp.ear</code>, that contains a module, <code>myejb.jar</code>:</p> <pre>java weblogic.Deployer -adminurl http://localhost:7001 -username weblogic -password weblogic -name myappdeployment -source ./myapp.ear -deploy</pre> <p>If you later try to target the EJB module to a new server using its package name, you will receive an error:</p> <pre>java weblogic.Deployer -adminurl http://localhost:7001 -username weblogic -password weblogic -name myappdeployment -targets myejb.jar@newserver -deploy</pre> <p>Application, <code>myappdeployment</code>, does not define the module, <code>myejb.jar</code>. Respecify the source to deploy additional modules for an application.</p> <p>This occurs because the <code>myejb.jar</code> module is named <code>myejb</code>, rather than <code>myejb.jar</code>.</p> <p><b>Workaround:</b> Deploy using exploded archive directories.</p>

## EJB Known Issues

Change Request Number	Description																								
CR061938	<p>In certain conditions, EJB QL queries can return spurious duplicates that are the results of SQL cross products. This can occur under the following conditions:</p> <ul style="list-style-type: none"> <li>• The EJB-QL query contains path expressions that navigate relationships; this generates multiple tables in the generated SQL SELECT clause.</li> <li>• The WHERE clause contains OR operands which navigate relationships, and not all of the path expression in the OR operands map to all of the tables in the generated SQL SELECT clause for the query; this may cause a cross product to show up in the results for that OR operand.</li> </ul> <p>The following example illustrates the problem:</p> <p><b>EJB QL:</b></p> <pre>SELECT OBJECT(c) FROM CustomerBean AS c, IN(c.accounts)accts WHERE c.name = '100' OR c.accts.bal = 300</pre> <p><b>DATA:</b> customer '100' exists but has no accounts</p> <p><b>EXPECTED RESULT:</b> customer '100' from clause #1</p> <p><b>ACTUAL RESULT:</b> customer '100' X number of accts</p> <p><b>SQL:</b></p> <pre>DROP TABLE thorick_customers; CREATE TABLE thorick_customers ( cust_name VARCHAR(10), cust_interests VARCHAR(10), cust_rating INTEGER, acct_id INTEGER, PRIMARY KEY (cust_name));  DROP TABLE thorick_accounts; CREATE TABLE thorick_accounts (acct_id INTEGER, bal FLOAT, PRIMARY KEY (acct_id));  INSERT INTO thorick_accounts VALUES (100, 100.0); INSERT INTO thorick_accounts VALUES (200, 200.0); INSERT INTO thorick_accounts VALUES (300, 300.0); INSERT INTO thorick_accounts VALUES (400, 400.0); INSERT INTO thorick_accounts VALUES (500, 500.0); INSERT INTO thorick_customers VALUES('100', 'jazz', 2, null); INSERT INTO thorick_customers VALUES('900', 'punk', 3, 400);  SELECT WL0.cust_name, WL0.cust_interests, WL0.acct_id FROM thorick_Customers WL0, thorick_Accounts WL1 WHERE WL0.cust_name = '100' OR ( w11.bal = 300 AND w10.acct_id=w11.acct_id );</pre> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">CUST_NAME</th> <th style="text-align: left;">CUST_INTER</th> <th style="text-align: left;">ACCT_ID</th> </tr> <tr> <th style="text-align: left;">-----</th> <th style="text-align: left;">-----</th> <th style="text-align: left;">-----</th> </tr> </thead> <tbody> <tr><td>100</td><td>jazz</td><td></td></tr> <tr><td>100</td><td>jazz</td><td></td></tr> <tr><td>100</td><td>jazz</td><td></td></tr> <tr><td>100</td><td>jazz</td><td></td></tr> <tr><td>100</td><td>jazz</td><td></td></tr> <tr><td>100</td><td>jazz</td><td></td></tr> </tbody> </table>	CUST_NAME	CUST_INTER	ACCT_ID	-----	-----	-----	100	jazz																
CUST_NAME	CUST_INTER	ACCT_ID																							
-----	-----	-----																							
100	jazz																								
100	jazz																								
100	jazz																								
100	jazz																								
100	jazz																								
100	jazz																								

---

<b>Change Request Number</b>	<b>Description</b>
CR093615	<p>There is a known issue with removal of stateful session beans in a cluster. Attempts to remove the bean with <code>EJBHome.remove</code> result in error messages such as this:</p> <pre data-bbox="357 491 1185 565">&lt;Dec 19, 2002 4:51:46 PM PST&gt; &lt;Info&gt; &lt;EJB&gt; &lt;BEA-010049&gt; &lt;EJB Exception in method : remove: java.rmi.NoSuchObjectException: Bean has been deleted.</pre> <p><b>Workaround:</b> Use <code>EJBObject.remove</code> instead of <code>EJBHome.remove</code> to remove a stateful session bean that is deployed in a cluster.</p>

---

## Examples Known Issues

Change Request Number	Description
CR068247	<p>The JSPs included with WebLogic Server were not being pre-compiled before they are accessed for the first time. If you were deploying precompiled JSPs, WebLogic Server may recompile them to the <code>/wlnotdelete/</code> directory.</p> <p><b>Workarounds:</b></p> <ol style="list-style-type: none"> <li>1. Precompile classes into your <code>WEB-INF/classes</code> directory (or a jar file in <code>WEB-INF/lib</code>) instead.</li> <li>2. Set a <code>workingDir</code> for the <code>jsp-descriptor</code> in your <code>weblogic.xml</code>: <pre data-bbox="431 756 1036 916"> &lt;jsp-descriptor&gt;   &lt;jsp-param&gt;     &lt;param-name&gt;workingDir&lt;/param-name&gt;     &lt;param-value&gt;d:\jsp_store&lt;/param-value&gt;   &lt;/jsp-param&gt; &lt;/jsp-descriptor&gt; </pre> </li> </ol>
CR099174	<p>When you configure new domains and servers in a WebLogic Server 8.1 installation, the MedRec Samples Authenticator is available as a provider for the security realm. However, this provider is only usable in a domain that includes the MedRec sample application and PointBase database.</p> <p>If you want to remove the MedRec Samples Authenticator from a newly-installed domain:</p> <ol style="list-style-type: none"> <li>1) Delete the <code>wlMedRecSampleAuthProvider.jar</code> file from the <code>WL_HOME\server\lib\mbeantypes</code> directory.</li> <li>2) In the Administration Console, go to the <code>Security-&gt;Realms-&gt;Providers-&gt;Authentication</code> node and delete the MedRec Samples Authenticator reference.</li> </ol> <p><b>Note:</b> If you follow these steps to remove the provider, you will not be able to log into the Patient Web Application of the MedRec sample application. The Patient application requires the MedRec Sample Authenticator to query the MedRec RDBMS for patient usernames, passwords, and assigned groups.</p>

Change Request Number	Description
CR101326	<p>The C++ clients for IIOP examples are not provided in this release. You will not be able to build or reference documentation for these clients until they are provided in a later release of the product, or on the <a href="http://dev2dev.bea.com">http://dev2dev.bea.com</a> site.</p> <p>This problem was resolved in 8.1 SP1.</p>
CR101713	<p>When running <code>examples.cluster.ejb</code> example, failover did not occur properly. Instead, the secondary server threw this exception:</p> <pre> Trouble while getting database connection java.sql.SQLException: Pool connect failed: java.lang.Exception: Weblogic Pool Driver doesn't support XA driver, Please change your config to use a Non-XA driver at weblogic.jdbc.common.internal.JDBCUtil.wrapAndThro wResourceException(JDBCUtil.java:161) at weblogic.jdbc.pool.Driver.connect(Driver.java:155) at weblogic.jdbc.jts.Driver.getNonTxConnection(Drive r.java:394) at weblogic.jdbc.jts.Driver.connect(Driver.java:137) at ..... </pre> <p>The error occurs because of an error in the sample instructions. <code>package.html</code> contained incorrect instructions for configuring the connection pool. The instructions say to use the default XA Pointbase pool. Instead, the pool should be configured as a non-XA pool. These instructions are corrected in WebLogic Server 8.1 SP2.</p>

## Installation Known Issues

Change Request Number	Description
CR101518	For the Japanese version of the installation, the shortcut to Online Documentation from the Windows Start menu linked to the English version of e-docs, rather than to the Japanese version of e-docs.

## Interoperability Known Issues

Change Request Number	Description
CR092181	Failover for entity beans and EJB handles requires that the cluster address be specified as a DNS name that maps to all server instances in the cluster and only server instances in the cluster. The cluster DNS name should not map to a server instance that is not a member of the cluster.
CR095479	Remote methods throwing <code>MarshalException/UnmarshalException</code> when interoperating between 7.0 SP1 and 8.1 servers may result in unmarshalling problems. This occurs because <code>MarshalException</code> and <code>UnmarshalException</code> map to the correct completion statuses in 8.1 but the incorrect completion statuses in 7.0 SP1.
CR100713	WebLogic Server 6.1 SP04 and earlier does not interoperate transactionally in some scenarios with WebLogic Server 8.1 over t3. Transactions between 6.1 and 8.1 server instances over t3 timeout with an <code>ArrayIndexOutOfBoundsException</code> on the 6.1 server instance.
CR101186	Exceptions thrown by WebLogic Server 8.1 cannot be decoded by WebLogic Server 7.0 SP02 or earlier. WebLogic Server 7.0 will throw a <code>MARSHAL</code> exception.
CR101324	Binding objects built with WebLogic Server 6.1 classes and deploying them to WebLogic Server 7.0 or higher is not supported and can result in failures during startup.

## WebLogic Server 8.1 Interoperability Notes

WebLogic Server 8.1 does not support mixed-version domains. All Managed Servers in a domain must be at the same version (8.1).

WebLogic Server 6.1 has some limitations when acting as a client to WebLogic Server 8.1 over the IIOP protocol. This is because WebLogic Server 6.1 does not support class evolution, so only identical classes can be passed between WebLogic Server 6.1 and WebLogic Server 8.1. These issues were partially resolved through the support of class evolution in WebLogic Server 7.0 and fully resolved in WebLogic Server 7.0 SP3. For WebLogic Server 7.0 IIOP clients interoperating with WebLogic Server 8.1, customers should use the WLS RMI-IIOP runtime by specifying `-Dweblogic.system.iiop.enableClient="true"` on the client command-line. This option is not supported in WebLogicServer 6.1 and not necessary in WebLogic Server 8.1.

In order to interoperate with WebLogic Server version 6.1, you must add the following elements to the `<Domain>` section of the WebLogic Server 8.1 `config.xml` file:

```
<!-- START OF 70 INTEROP SPECIFIC SETTING -->
<Security CompatibilityMode="true" GuestDisabled="false"
  InteropEnabled="true"
  InteropPassword="{3DES}OuQN48TIXRYFVVeZ4VBFTA=="
  InteropUsername="system" Name="@DOMAIN1"
  PasswordPolicy="wl_default_password_policy" Realm="wl_default_realm"
  RealmSetup="true" SystemUser="system"/>
<!-- START PER CR071185, ADDING BELOW SETTING TO 70 ONLY -->
<SecurityConfiguration Credential="gumby1234" Name="@DOMAIN1" />
<!-- END PER CR071185, ADDING BELOW SETTING TO 70 ONLY -->
<!-- END OF 70 INTEROP SPECIFIC SETTING -->
```

(The above elements are also required for WebLogic Server version 7.0 to interoperate with WebLogic Server version 6.1.)

## JVM Interoperability

When WebLogic Server instances interoperate with other WebLogic Server instances, each server instance must use a certified JVM. For example, to interoperate with WebLogic Server 8.1, WebLogic Server versions 6.1 and 7.0 are certified to use Sun's JDK version 1.3.1\_x, while WebLogic Server 8.1 is certified to use JDK version 1.4.1\_02. See [Supported Configurations](http://e-docs.bea.com/wls/certifications/certifications/index.html) at <http://e-docs.bea.com/wls/certifications/certifications/index.html> for more information on certified JVMs.

For standalone clients (rather than server instances) the same JVM considerations apply.

WebLogic Server 8.1 running on JDK 1.4.1 will interoperate with 7.0 SP2 running on JDK 1.3.1.

## J2EE Known Issues

Change Request Number	Description
CR076513	<p>You cannot deploy a standalone Enterprise Application module (Web Application, EJB, or Resource Adapter) when an unrelated META-INF/application.xml file resides in the same directory. For example, you cannot deploy a .rar file from c:\mydeployment\myadapter.rar if an unrelated c:\mydeployment\META-INF\application.xml file exists.</p> <p><b>Workaround:</b> Delete the unrelated META-INF directory or place the module in a dedicated directory.</p>

## JDBC Known Issues

Change Request Number	Description
CR177621	<p>PointBase Server is an all-Java DBMS product included in the WebLogic Server distribution solely in support of WebLogic Server evaluation, either in the form of custom trial applications or through packaged sample applications provided with WebLogic Server. Non-evaluation development and/or production use of the PointBase Server requires a separate license to be obtained by the end user directly from PointBase.</p>
n/a	<p>The connection creation retries option for a connection pool depends on the message returned from the JDBC driver to determine whether to continue to try to create database connections. In some cases, Weblogic Server will not retry to create database connections.</p> <p><b>Workaround:</b> Correct the problem with your database, and then untarget and retarget the connection pool in the Administration Console. See <a href="#">“Deploying a JDBC Connection Pool to One or More Servers or Clusters”</a> in the <i>Administration Console Online Help</i>.</p> <p>For more information about the connection creation retries option, see the <a href="#">“Connection Creation Retry Frequency”</a> page in the <i>Administration Console Online Help</i> and <a href="#">“Enabling Connection Creation Retries”</a> in <i>Programming WebLogic JDBC</i>.</p>

Change Request Number	Description
n/a	<p>The JDBC Connection Pool Assistant in the Administration Console automatically sets the Test Table Name attribute for a connection pool based on the DBMS of the JDBC driver that you select. The Test Table Name attribute is used in connection testing which is optionally performed periodically or when you create, reserve, or release a connection, depending on how you configure the connection pool. For the test to succeed, the database user used to create database connections in the connection pool must have access to the database table. If not, you should either grant access to the user (make this change in the DBMS) or change the Test Table Name attribute to the name of a table to which the user does have access (make this change in the WebLogic Server Administration Console). For a list of default values, see <a href="#">“Default Test Table Name”</a> in the <i>Administration Console Online Help</i>.</p>
CR082336	<p>In WebLogic Server 8.1, dbKona was removed from the distribution. It was deprecated in previous releases.</p>
CR086874	<p>Processes can no longer share a reference to a connection from a connection pool. For example, a JSP cannot request and receive a connection from an EJB. This was allowed in previous releases of WebLogic Server, but is no longer allowed because it occasionally resulted in multiple threads accessing a single connection or invalid references because a connection was closed or garbage collected.</p>
CR086902	<p>For connection pools that connect to a PointBase database, statement caching is disabled by default because PointBase does not effectively handle cached prepared or callable statements after a table has been dropped and recreated. If your code does not include such DDL operations, you can enable statement caching in the Administration Console. See <a href="#">“Increasing Performance with the Statement Cache”</a> page in the <i>Administration Console Online Help</i>.</p>
CR093175	<p>The <code>JDBCDataSourceMBean</code> is deprecated in WebLogic Server 8.1. Use the <code>JDBCTxDataSourceMBean</code> instead. The attributes that are not available in the <code>JDBCTxDataSourceMBean</code> (<code>WaitForConnectionEnabled</code> and <code>ConnectionWaitPeriod</code>) have been deprecated and are replaced with the <code>ConnectionReserveTimeoutSeconds</code> attribute in the <code>JDBCConnectionPoolMBean</code>.</p>

Change Request Number	Description
CR099442	<p>The documentation regarding Emulate Two Phase Commit in the Administration Console Online Help as installed with the WebLogic Server software is incorrect. This feature does not use the last-agent commit optimization, as documented. The correct documentation is available in the e-docs version of the <a href="#">Administration Console Online Help</a>.</p>
CR100625	<p>Several weblogic.Admin JDBC commands behave differently than documented in “<a href="#">Commands for Managing JDBC Connection Pools</a>” in the <i>WebLogic Server Command Reference</i>. The documentation describes the desired behavior, rather than the actual behavior. The commands will be fixed in a future release to match the documentation. Commands with behavior other than what is documented:</p> <ul style="list-style-type: none"> <li data-bbox="391 739 1065 947"> <p>• <b>SUSPEND_POOL</b> - This command does not take true or false parameters. Instead, it defaults to the behavior for the false option. Syntax:</p> <pre data-bbox="431 835 1042 947">java [SSL trust options] weblogic.Admin [ [-url   -adminurl] URL] -username username -password password SUSPEND_POOL -poolName connection_pool_name</pre> </li> <li data-bbox="391 956 1065 1164"> <p>• <b>SHUTDOWN_POOL</b> - This command does not take true or false parameters. Instead, it defaults to the behavior for the false option. Syntax:</p> <pre data-bbox="431 1052 1042 1164">java [SSL trust options] weblogic.Admin [ [-url   -adminurl] URL] -username username -password password SHUTDOWN_POOL -poolName connection_pool_name</pre> </li> </ul>
CR101419	<p>Fail-over for DataSources and TxDataSources in a cluster is not working in this release. This problem is resolved in Service Pack 1.</p>

## JMS Known Issues

Change Request Number	Description
CR097038	<p>When a source JMS bridge destination is not running in the same WebLogic Server 8.1 domain as the Message Bridge, then a trusted security relationship must be established between the domains. Otherwise, the Messaging Bridge must be configured to work in synchronous mode (Asynchronous Mode Enabled = No).</p> <p>For more information on configuring security interoperability for the Messaging Bridge, see “<a href="#">Messaging Bridge</a>” in the <i>Administration Console Online Help</i>.</p>
CR098975	<p>WebLogic Server sometimes fails to boot after you undeploy a distributed destination. This occurs when you delete the distributed members and the distributed destination itself, but do not delete the underlying physical destinations. The following sequence of Administration Console actions will cause this problem:</p> <ol style="list-style-type: none"> <li>1. In the Configuration -&gt; Members tab of a distributed destination, click the trash can icon to remove a member.</li> <li>2. When prompted, do not delete the member’s underlying physical destination.</li> <li>3. Click Remove to delete the member.</li> <li>4. If necessary, continue deleting all the members of the distributed destination.</li> <li>5. Delete the distributed destination.</li> </ol> <p>If you shut down the server at this point, you will not be able to reboot it.</p> <p><b>Workaround:</b> Delete each distributed destination member’s underlying physical destination when prompted by the Administration Console.</p>

## JVM Known Issues

---

Change Request Number	Description
CR099314	<p data-bbox="391 461 1069 661">There is a problem with JDK 1.3.1_04 and the nohup command, as described at <a href="http://developer.java.sun.com/developer/bugParade/bugs/4755829.html">http://developer.java.sun.com/developer/bugParade/bugs/4755829.html</a>. The nohup command starts failing; the WebLogic Server process exits when the parent shell exits, regardless of the invocation of nohup. The problem was confirmed for ksh (the default Solaris 8 shell), but may exist for other shells. The problem does not occur with the bash shell.</p> <p data-bbox="391 678 525 699"><b>Workarounds:</b></p> <ul data-bbox="391 716 1069 835" style="list-style-type: none"><li data-bbox="391 716 1069 772">• <a href="http://askbea-int.beasys.com/askbea/wls/S-15924.html">http://askbea-int.beasys.com/askbea/wls/S-15924.html</a> provides a workaround (the <code>-Xrs</code> flag), but at the cost of losing thread dumps.</li><li data-bbox="391 782 1069 835">• Use “(startWeblogic &gt;output 2&gt;&amp;1 &amp;)” instead of “nohup startWeblogic &amp;”.</li></ul>

---

## Oracle Thin Driver Known Issues

**Note:** In addition to the issues listed below, see [“Known Oracle Thin Driver Issues”](#) in *Programming WebLogic JTA*. That section contains information about known problems in the Oracle Thin Driver 9.2.0, and ways to work around the problems. WebLogic Server 8.1 ships with the 9.2.0.2 version of the Oracle Thin Driver, which corrects some known issues not listed here or in *Programming WebLogic JTA*.

Change Request Number	Description
n/a	<p>After transaction recovery has completed from a server restart, some customers have seen frequent occurrences of the ORA-1591 exception, which may be followed by ORA-24756 or ORA-24776 exceptions.</p> <p><b>Workarounds:</b></p> <p>If you see these errors, you should refer to the following Oracle notes: Note:100664.1 and Doc ID: 231196.1.</p> <p>You should also ensure that the application is not leaking connections from the JDBC connection pool. You can check by:</p> <ol style="list-style-type: none"> <li>1. Enabling Connection Leak Profiling on the JDBC Connection Pool --&gt; Configuration --&gt; Connections tab in the Administration Console (under Advanced Options).</li> <li>2. In the navigation tree in the Administration Console, right-click the connection pool used in the application and select View Leaked Connections.</li> </ol> <p>For more information, see <a href="#">“Enable Connection Leak Profiling”</a> in the <i>Administration Console Online Help</i>.</p> <p>If you continue to see ORA-1591 errors, you may want to contact Oracle Support for help with tuning your database. Before contacting Oracle Support, consider doing the following:</p> <ul style="list-style-type: none"> <li>• Generate a trace file for the ORA-1591.</li> <li>• Run the statspack report and analyze contentions in the database.</li> </ul>

---

<b>Change Request Number</b>	<b>Description</b>
CR101709	<p data-bbox="391 388 1067 527">Application code can inadvertently start local transactions on a Connection object, which can cause XAER_PROTO errors. According to the SQL spec, any DDL/DML operation can potentially start a local transaction. It is the application's responsibility to commit or rollback these transactions.</p> <p data-bbox="391 545 1067 774">This was not a problem with earlier versions of the Oracle Thin driver because earlier versions of the driver did not check for active local transactions before starting a global transaction or active global transactions before starting a local transaction. However, version 920 of the Oracle Thin driver has started checking for this. It throws an XAER_PROTO error if an application (or WebLogic Server) tries to start a global transaction (<code>XAResource.start()</code>) on a connection on which a local transaction is already active.</p> <p data-bbox="391 791 518 817"><b>Workaround:</b></p> <p data-bbox="391 835 1067 913">The application code must end the local transaction as an additional step before closing the connection (which returns the connection to the connection pool).</p> <p data-bbox="391 930 733 956">This problem was resolved in 8.1 SP1.</p>

---

## RMI/RMI-IIOP Known Issues

<b>Change Request Number</b>	<b>Description</b>
CR095810	<p>You cannot use callbacks through a firewall when using the thin client .jar files (<code>wlclient.jar</code> and <code>wljmsclient.jar</code>) and IIOP or IIOPS.</p> <p><b>Workaround:</b> Callbacks do work when using tunneling.</p> <p>This problem was resolved in 8.1 SP1.</p>
CR101476	<p>An applet using the thin client .jar files (<code>wlclient.jar</code> and <code>wljmsclient.jar</code>) cannot perform tunneling using the Sun Java2 SDK 1.4.1_02 installed with WebLogic Server. This problem is fixed in Sun's SDK versions 1.4.1_03 and 1.4.2.</p>

Change Request Number	Description
CR124265	<p>WebLogic Server 8.1 provides its own ORB implementation in place of the ORB shipped with J2SE 1.4. This ORB is instantiated by default when programs call <code>ORB.init()</code>, or when "java:comp/ORB" is looked up in JNDI. Customers who wish to use a different ORB by default should set the following properties:</p> <pre>org.omg.CORBA.ORBSingletonClass=&lt;classname&gt; org.omg.CORBA.ORBClass=&lt;classname&gt;</pre> <p>Note that <code>ORBSingletonClass</code> must be set on the server command-line, whereas <code>ORBClass</code> can be set as a property argument to <code>ORB.init()</code>. Customers wishing to use a different RMI-IIOP implementation must set the following two properties:</p> <pre>javax.rmi.CORBA.UtilClass=&lt;classname&gt; javax.rmi.CORBA.PortableRemoteObjectClass=&lt;classname&gt;</pre> <p>Customers who do this will get the following errors at server startup:</p> <pre>&lt;Sep 19, 2003 9:12:03 AM CDT&gt; &lt;Error&gt; &lt;IIOP&gt; &lt;BEA-002015&gt; &lt;Using javax.rmi.CORBA.UtilClass &lt;classname&gt;; The IIOP subsystem requires a WebLogic Server-compatible UtilClass.&gt; &lt;Sep 19, 2003 9:12:03 AM CDT&gt; &lt;Error&gt; &lt;IIOP&gt; &lt;BEA-002016&gt; &lt;Using javax.rmi.CORBA.PortableRemoteObjectClass &lt;classname&gt;, the IIOP subsystem requires a WebLogic Server-compatible PortableRemoteObjectClass.&gt;</pre> <p>indicating that the WebLogic RMI-IIOP runtime will not work.</p> <p>The J2SE defaults for these properties are:</p> <pre>org.omg.CORBA.ORBSingletonClass=com.sun.corba.se. internal.corba.ORBSingleton org.omg.CORBA.ORBClass=com.sun.corba.se.internal. Interceptors.PIORB javax.rmi.CORBA.UtilClass=com.sun.corba.se.intern al.POA.ShutdownUtilDelegate javax.rmi.CORBA.PortableRemoteObjectClass=com.sun .corba.se.internal.javax.rmi.PortableRemoteObject</pre> <p>This problem was resolved in 8.1 SP2.</p>

---

<b>Change Request Number</b>	<b>Description</b>
CR186728	JMS IIOP thin client fails with the following exception: <code>JMSServer "add JMS Dispatcher peerGone". Unhandled exception.</code> For more information, see <a href="#">Setting Network Channel Addresses</a> .

---

## Security Known Issues

For information about BEA security advisories, refer to the BEA Advisories & Notifications page on the dev2dev Web site. On this page, you can download security-related patches and register to receive notifications of newly available security advisories. You can access the Advisories & Notifications page with the following URL: <http://dev2dev.bea.com/advisories>.

BEA has established an e-mail address ([security-report@bea.com](mailto:security-report@bea.com)) to which you can send reports of any possible security issues in BEA products.

The following table summarizes security-related known issues for the 8.1 release.

Change Request Number	Description
CR090469	<p>If you are upgrading from WebLogic Server 6.1 or earlier and you want to use the security features that were introduced in WebLogic Server 7.0, your domain's <code>config.xml</code> file must declare a configuration version of 7.0 or later. For example,</p> <pre data-bbox="348 604 870 652">&lt;Domain ConfigurationVersion="8.1.0.0" Name="mydomain"&gt;</pre> <p>If you are using a <code>config.xml</code> file that you created with WebLogic Server 6.1 or earlier, you must open <code>config.xml</code> in a text editor and add <code>ConfigurationVersion="8.1.0.0"</code> to the <code>&lt;Domain&gt;</code> element.</p>
CR091648 CR099358	<p>The MedRec Authentication provider implements authentication for the MedRec sample applications and cannot be used in a production domain.</p> <p><b>Workaround:</b></p> <p>Delete the JAR file for the MedRec Authentication provider (<code>wlMedRecSampleProvider</code>) from the <code>WL_HOME\lib\mbeantypes</code> directory and reboot WebLogic Server. The MedRec Authentication provider should no longer appear in the list of available Authentication providers on the Providers--&gt;Authentication Providers tab in the WebLogic Server Administration Console.</p>
CR097038	<p>When using the Messaging Bridge, non-trusted domains will only work when using synchronous mode communications.</p> <p><b>Workaround:</b> WebLogic Server only supports interoperability between domains for synchronous operations.</p>

Change Request Number	Description
CR100783	<p>The following exception may occur if the password to unlock the private key was entered incorrectly:</p> <pre data-bbox="411 491 1130 881"> ImportPrivateKey failed, java.security.KeyManagementException:ASN.1: Lengths longer than 32 bits are not supported java.security.KeyManagementException:ASN.1: Lengths longer than 32 bits are not supported at com.certicom.security.cert.internal.x509.SSLPlusSupport. getLocalIdentityPartial(Unknown Source) at com.certicom.net.ssl.CerticomContextWrapper. inputPrivateKey(Unknown Source) at utils.ImportPrivateKey.importKey(ImportPrivateKey.java:67) at utils.ImportPrivateKey.main(ImportPrivateKey.java:23) </pre> <p>It is also possible to get this exception if the private key was corrupt or was some format that WebLogic Server does not recognize.</p> <p><b>Workaround:</b> Re-enter the correct password. If you have forgotten the password for the private key or if the private key is corrupt, obtain a new private key.</p> <p>This problem was resolved in 8.1 SP1.</p>
CR101274	<p>Please review the security advisory information at <a href="http://www.ncipher.com/support/advisories">http://www.ncipher.com/support/advisories</a>.</p> <p>This problem was resolved in 8.1 SP1.</p>

<b>Change Request Number</b>	<b>Description</b>
CR101299	<p>Figure 4-7, Auditing Process, in the <i>Introduction to WebLogic Security</i> document is incomplete in as much as it does not show all the Audit events generated by WebLogic Server.</p> <p>Not only are Audit events written for authentication success and failure (when identity is authenticated, impersonated, and asserted), Audit events are also written for Authorization provider exceptions, Adjudication provider decisions, user lockout unlock, locked, and lockout expired actions, Role Mapping provider exceptions, and when security roles are obtained.</p> <p>Figure 4-7 is correct but incomplete in that it only refers to authentication and does not imply that an authentication action can result in different audit events depending upon what happens.</p>
CR101770	<p>The SSL Login Timeout attribute was unintentionally dropped from the WebLogic Server Administration Console. Support for the attribute was added in WebLogic Server 8.1 SP1.</p> <p>This problem was resolved in 8.1 SP1.</p>

## Servlets & JSPs Known Issues

Change Request Number	Description
CR100068 CR110324	<p>You can not use JSTL tags in JSPs that include Japanese characters. When such a JSP is executed, an error starting with the following lines occurs:</p> <pre>java.io.IOException: javax.servlet.jsp.JspException: The taglib validator rejected the page: "org.xml.sax.SAXParseException: An invalid XML character (Unicode: 0x82) was found in the CDATA section." ...</pre> <p>This problem occurs because of the following, conflicting conditions:</p> <ol style="list-style-type: none"> <li>1. The page encoding of JSPs is defined by <code>Shift_JIS</code>, as in:  <code>&lt;%@ page pageEncoding="Shift_JIS" %&gt;</code></li> <li>2. Using multibyte characters (Japanese) in JSPs.</li> <li>3. Using JSTL tags.</li> </ol>

## System Administration Known Issues

Change Request Number	Description
CR085065	<p>The <code>WebLogicMBean.setName</code> method, which is inherited by all WebLogic Server MBeans, is no longer supported. The method changes the <code>Name</code> component of an MBean instance's <code>WebLogicObjectName</code>. Because this component is part of the primary key for a <code>WebLogicObjectName</code>, changing the value leads to undesired behavior.</p> <p>Instead of setting the value of an MBean's <code>Name</code> attribute, the API throws <code>InvalidAttributeValueException</code>.</p> <p><b>Workaround:</b> If you want to change the name of an MBean instance, destroy the current instance and create a new one with a new name.</p>
CR094188	<p>If you place an application in the <code>servername/applications</code> directory and then start the server in development mode, the server automatically deploys the application, but it does not save the application configuration to the <code>config.xml</code> file. If you start the server in production mode, the server neither deploys the application nor saves the configuration to the <code>config.xml</code> file.</p> <p><b>Workaround:</b> Start the server first and then deploy the application. In development mode, you can deploy the application by copying it to the <code>servername/applications</code> directory.</p>
CR100089	<p>Within a domain, each server, machine, cluster, JDBC connection pool, virtual host, and any other resource type must be named uniquely and must not use the same name as the domain.</p> <p>If you create a resource with a non-unique name, WebLogic Server emits an error message.</p> <p>This requirement applies to the <code>Name</code> component of <code>WebLogicObjectName</code> as well. For example, if you use WebLogic Server APIs to create Configuration MBeans, the value of the <code>Name</code> component must be unique amongst all other MBeans in the domain.</p>

## Tools Known Issues

Change Request Number	Description
CR097668	<p>When using cygwin version 1.3.20(0.73/3/2) the shell scripts fail if the cygdrive prefix has been changed (for example, mount -c /so /c/windows instead of /cygdrive/c/windows). This is because the value '/cygdrive' is hardcoded in the scripts. The following scripts are affected:</p> <pre>wblogic700/common/bin/commEnv.sh wblogic700/server/bin/setWLSEnv.sh wblogic700/server/bin/startNodeManager.sh wblogic700/server/bin/startWLS.sh</pre> <p><b>Workaround:</b> Replace the value of /cygdrive in these scripts to get the correct cygdrive prefix:</p> <pre>CYGDRIVE=`mount -ps   tail -1   awk '{print \$1}'   sed -e 's%/%%'`</pre> <p>For example, in startWLS.sh find:</p> <pre>if [ -n "`uname -s  grep -i cygwin_`" ]; then   WL_HOME=`echo \$WL_HOME   sed "s#\([a-zA-Z]\):#/cygdrive/\1#g"`   JAVA_HOME=`echo \$JAVA_HOME   sed "s#\([a-zA-Z]\):#/cygdrive/\1#g"` fi</pre> <p>and change it to:</p> <pre>if [ -n "`uname -s  grep -i cygwin_`" ]; then   CYGDRIVE=`mount -ps   tail -1   awk '{print \$1}'   sed -e 's%/%%'`   WL_HOME=`echo \$WL_HOME   sed "s#\([a-zA-Z]\):#{CYGDRIVE}/\1#g"`   JAVA_HOME=`echo \$JAVA_HOME   sed "s#\([a-zA-Z]\):#{CYGDRIVE}/\1#g"` fi</pre> <p>Change the other scripts in a similar manner.</p>

## WebLogic jDriver for Oracle Known Issues

---

<b>Change Request Number</b>	<b>Description</b>
n/a	You cannot use <code>windows-31j</code> encoding with the BEA WebLogic JDriver™ for Oracle because of a known limitation in the Sun Java 2 SDK 1.4.1 (Bug Id 4715330): the <code>windows-31j</code> converter is missing in <code>lib/charsets.jar</code> . The encoding of WebLogic jDriver for Oracle is based on <code>lib/charsets.jar</code> , so this character encoding is not supported on the Sun Java 2 SDK 1.4.1. Sun intends to fix this problem in the Java 2 SDK 1.4.1_05 and 1.4.2 release. BEA fixed this problem in the WebLogic JRockit 8.1 Service Pack 1.

---

## WebLogic Tuxedo Connector Known Issues

Change Request Number	Description
CR067275	Load balancing and failover is not supported with WebLogic Tuxedo Connector in a cluster using RMI over IIOP.
CR079630	<p>Most WebLogic Tuxedo Connector tBridge users will expect to set the Correlation ID using the <code>setJMSCorrelationID(String)</code> method. This will take a 32 character string and turn it into a 64 byte array. JMS stores the string as UTF-16BE. When the tBridge receives the Correlation ID from Tuxedo it is 32 characters as 32 bytes. The tBridge then uses the <code>setJMSCorrelationIDAsBytes(byte[])</code> method to set the message for JMS receiving queue. The characters may be the same in ASCII but the two will not compare due the length difference.</p> <p><b>Workaround:</b></p> <p>When it is necessary to compare the Correlation ID with the ID returned from Tuxedo in the receiving JMS queue, create a byte array containing the hexadecimal values of the Correlation ID. Then use the <code>setJMSCorrelationIDAsBytes()</code> and <code>getJMSCorrelationIDAsBytes()</code> methods to create an ID to compare with the ID returned from Tuxedo in the receiving JMS queue.</p> <p>For example, if the string Correlation ID is  <code>"1234567890ABCDEFGHIJKLMNQRSTUUV"</code>:</p> <pre>private byte[] coridbyte={0x31,0x32,0x33,0x34,0x35,0x36,0x37,                           0x38,0x39,0x30,0x41,0x42,0x43,0x44,                           0x45,0x46,0x47,0x48,0x49,0x4a,0x4b,                           0x4c,0x4d,0x4e,0x4f,0x50,0x51,0x52,                           0x53,0x54,0x55,0x56}; msg.setJMSCorrelationIDAsBytes(coridbyte); corIDasBytes = msg.getJMSCorrelationIDAsBytes(); corIDasBytes contains the correct value to compare with the Correlation ID returned from Tuxedo.</pre>

---

<b>Change Request Number</b>	<b>Description</b>
CR095050	When using the WTCMigrateCF utility to migrate WebLogic Server 6.1 WTC configuration files, Weblogic Server throws NameNotFoundException when using the -deploy option. Users should ignore this exception in the server log.
CR095807	Linux and AIX compilers may produce compile errors when using Tuxedo based IDL with some value types. This issue is resolved in Tuxedo 8.1 RP001.

---

## WebLogic Workshop Known Issues

Change Request Number	Description
n/a	WebLogic Server 8.1 ships with a Beta version of the WebLogic Workshop IDE and development framework. In the Beta release, WebLogic Workshop requires 1GB of system RAM.
n/a	WebLogic Workshop is not available in the Japanese version of WebLogic Server 8.1.
CR097002	WebLogic Server 8.1 ships with two JSP compilers: the original WebLogic JSP compiler, and a compiler that is optimized for use with WebLogic Workshop. Internationalization for the WebLogic Workshop-optimized JSP compiler will be completed after the initial release of WebLogic Server 8.1.
CR102094	The script used to start the WebLogic Workshop sample server, <code>WL_HOME\samples\workshop\startWebLogic.cmd</code> is not installed in this release. You cannot use the Windows Start Menu selection to start the sample server because it refers to this missing file.

## Web Services Known Issues

Change Request Number	Description
CR063817	A complex type which contains an element of type <code>xsd:schema</code> causes an exception during autotyping.
CR074740	WebLogic Server does not convert Java data structures which contain Lists as items of Lists to an XML Schema.
CR083527	The <code>typeMappingBase</code> attribute of autotyper is ignored when generating holder classes.
CR086326	The <code>-Dweblogic.webservice.verbose=true</code> does not work for SSL clients. The workaround to get verbose output is to use <code>-Dweblogic.webservice.binding.verbose=true</code> .

Change Request Number	Description
CR092184	A service configured to use the security extensions cannot be accessed from the Web service test page.
CR101160	<p>Complex data types in service-specific exceptions are not handled in strict accordance with the Java API for XML-Based RPC (JAX-RPC) 1.0 specification, as follows:</p> <ul style="list-style-type: none"> <li> <b>Section 4.3.6 WSDL Fault, <i>Service Specific Exception</i>:</b> When the <code>clientgenAnt</code> task generates the JAX-RPC stubs from a WSDL that contains a <code>wSDL:fault</code> element whose message part is an <code>xsd:complexType</code>, the generated stubs incorrectly map the <code>xsd:complexType</code> to a simple Java type. This means that when a client application uses the stubs to invoke the Web Service, and the service throws the service-specific exception (rather than the <code>javax.xml.rpc.SOAPFaultException</code>), the client application cannot access the full complex data from the exception. There is no workaround to this problem.                 </li> <li> <b>Section 5.5.5 Methods, <i>Exceptions</i>, fourth bullet:</b> The WebLogic Web Service runtime incorrectly maps multiple fields in a service-specific Java exception class to a simple XML data type, rather than to an <code>xsd:complexType</code>.                 </li> </ul> <p><b>Workaround:</b> Always throw a <code>javax.xml.rpc.SOAPFaultException</code>, rather than a service-specific exception, when implementing a WebLogic Web Service.</p>

## XML Known Issues

Change Request Number	Description
CR102744	Although the class <code>weblogic.xml.xpath.XMLNodeXPath</code> is not public, its Javadoc was mistakenly exposed. BEA does not support the use of this internal class.

# Resolved Problems for Service Pack 5

Service Packs are cumulative; Service Pack 5 contains all fixes made in earlier Service Packs released for BEA WebLogic<sup>®</sup> Server 8.1.

The following sections describe problems that were resolved in BEA WebLogic Server<sup>®</sup> 8.1 Service Pack 5:

- [“Administration Console” on page 3-3](#)
- [“Cluster” on page 3-8](#)
- [“Connector” on page 3-8](#)
- [“CORBA” on page 3-9](#)
- [“Core WebLogic Server” on page 3-10](#)
- [“Deployment” on page 3-15](#)
- [“EJB” on page 3-15](#)
- [“Installer” on page 3-23](#)
- [“J2EE” on page 3-23](#)
- [“jCOM” on page 3-23](#)
- [“JDBC” on page 3-24](#)
- [“jDriver” on page 3-30](#)

## Resolved Problems for Service Pack 5

- “JMS” on page 3-30
- “JNDI” on page 3-35
- “JTA” on page 3-35
- “JVM” on page 3-37
- “Node Manager” on page 3-37
- “Operations, Administration, and Management” on page 3-39
- “Plug-Ins” on page 3-47
- “RMI” on page 3-51
- “Security” on page 3-53
- “Servlets and JSPs” on page 3-57
- “SNMP” on page 3-63
- “Tools” on page 3-64
- “Utilities” on page 3-65
- “Web Services” on page 3-65
- “WebLogic Tuxedo Connector” on page 3-70
- “XML” on page 3-72

# Administration Console

<b>Change Request Number</b>	<b>Description</b>
CR081329	The Refresh button no longer opens the page that was previously opened, but instead refreshes the page that is currently opened.
CR106556	No exception was being thrown when an existing provider with the same name was replaced. Now, WebLogic Server throws an InstanceAlreadyExistsException if an MBean is already registered with the given name. The error message is displayed in the Administration Console.
CR107353	The Administration Console only displays the 48 users per authentication provider in a security realm. A filter was provided to narrow the results, but there was no indication that there were more users than what was being displayed.  Now, the console indicates to the user that there are more users in the security realm than the console can display.
CR107977	The Administration Console in Japanese mode now displays the expected content when you select Servers in the left pane and then click Customize this view... in the right pane, regardless of the browser you are using.
CR135899	If the Administration Server and the Managed Server are running on different multi-byte locales, the Managed Server log can now be viewed correctly in the Administration Console when you select the locale of the Managed Server in the Console Preferences page.
CR174743	WebLogic Server correctly updates the <code>config.xml</code> file when the Administration Console is used to change the load order of Web applications. Previously, the leading “/” character in the application path was incorrectly overwritten.
CR178121	Correct values of user lockout attributes are now shown in the Administration Console with custom security providers in non-default security realms other than the default security realm.

Change Request Number	Description
CR178658	<p>The Servlet Extension Case Sensitive attribute has been added to server and cluster configurations, and the Web App Files Case Insensitive attribute has been added to the security domain configuration.</p> <p>These attributes specify whether file lookups for Java Server Pages (JSPs) are case sensitive on all platforms except win32; file lookups from standard win32 file systems are always case-insensitive. On case-insensitive file systems other than win32 (such as NT Samba mounts from UNIX or Mac OS that have been installed in case-insensitive mode), specify case-sensitive lookups by setting these attributes to False to prevent the JSP from returning its source code. For example, if a JSP is being served from a Samba mount and you have specified case-insensitive lookups, WebLogic Server converts all file name extensions to lower case before looking up the JSP.</p> <p>Possible values and usage:</p> <p>OS (the default value): WebLogic Server relies on the operating system for pattern matching.</p> <p>True: WebLogic Server enforces case-insensitivity irrespective of the operating system.</p> <p>False: WebLogic Server enforces case-sensitivity irrespective of the operating system.</p>
CR180077	<p>The custom attribute and the Display Message appear in the Administration Console under the Details tab. Help text for security provider custom attributes is displayed in the description tag of the attribute defined in the descriptor files.</p>
CR183868	<p>When deleting an application from the server, the message in the Administration Console no longer indicates that the deletion process has been completed. Instead, now the message indicates that the deletion process will be completed.</p>
CR191904 CR215149	<p>New servers can be added to a cluster without error even if the cluster has an existing list of members that are running.</p>
CR191989	<p>When you clone a server through the Administration Console, the <code>LogMBean</code> and the <code>ExecuteQueues</code> are now properly copied.</p>
CR196590	<p>The Administration Console now shows Redirections as the child of the WTC Service itself instead of as the child of <code>tBridge</code>.</p> <p>Also, the Administration Console no longer allows you to configure identical Redirections (that is, Redirections with the same name) to the same WTC Service.</p>
CR199870	<p>If an error occurs while you are configuring a new execute thread, the error is now displayed on the current page rather than on the MBean parent's page.</p>

<b>Change Request Number</b>	<b>Description</b>
CR199916	The Virtual Machine no longer crashes during redeployment.
CR234184	Synchronization was added to the deployment descriptor editor code so that only a single thread can un-jar the application descriptor.
CR201904	When you use the Target Each Module button to deploy an EAR file containing one or more Web applications, the deployment no longer fails.
CR202739	The Apply button is now displayed in the Administration Console for Monitor, Operator, and Deployer IDs and in the Customize Logs page.
CR206190	The Administration Console threw an error if the Console extension did not contain <code>index.xml</code> and the catalog for the selected language or locale.  The Administration Console uses the default catalog if <code>index.xml</code> is not present or if no catalog is defined for the selected language or locale.
CR208509	You can now use the DISCOVERMANAGEDSERVER button, to re-establish administrative control over Managed Servers even after the Administration Server has already started, using the Administration Console.
CR208547	The inline help text has been updated to show that the custom Hostname Verifier can implement either <code>weblogic.security.SSL.HostnameVerifier</code> (deprecated) or <code>weblogic.security.SSL.HostnameVerifierJSSE</code> .
CR209781	The Administration Console left pane (applet) is now working properly with the Microsoft JVM.
CR211026	When an EJB was deployed on a Managed Server and the in-flight transaction was monitored, the transaction appeared on the Administration Console, but could not be rolled back by using the <code>Force Local Rollback</code> and <code>Force Global Rollback</code> functions.  The <code>Force Local Rollback</code> and <code>Force Global Rollback</code> now work properly.
CR212870	When two applications with one application-scoped datasource per application were deployed on the same server and the datasources had the same name, an <code>InstanceAlreadyExistsException</code> was thrown.
CR126808	
CR215833	Code was added to configure the pool name to be application-specific when necessary.
CR214041	Opening the Global Role editor pop-up in the Administration Console no longer results in an Apache 500 error.

Change Request Number	Description
CR215112 CR221786	<p>The following pages in the Administration Console are now consistent in how they show the current state of Managed Servers:</p> <ul style="list-style-type: none"> <li>• Domain Monitoring</li> <li>• Domain Control</li> <li>• Servers Table</li> <li>• Server Monitoring</li> <li>• Server Control</li> </ul>
CR215352	Force Local/Global Commit now works against the transaction running on Managed Servers.
CR216670	When you exceed your number of authentication attempts, a message is logged throughout the domain saying that you are locked out. Now, the Administration Console also shows the lockout message.
CR216851	The Define Security Policy and the Define Scoped Role pop-up menu options are no longer available for JMS bridge destinations.
CR217508	The banner in the Administration Console now shows the user authenticated against WebLogic Server instead of the proxy user ID.
CR217812	The Administration Console Web Application deployment testing tab can now handle the default <code>context (/)</code> . Now, no extra <code>/</code> will be added with the default <code>context (/)</code> .
CR222458	<p>Clicking the Administration Console deploy button restaged Web Applications to targeted servers.</p> <p>The Administration Console no longer restages files that already exist.</p>
CR222872	The order of filter connection rules displayed in the Administration Console is now the same as the order you entered before you click the Apply button.
CR223073	<p>If the method signature contained array as parameter, the method name was not passed to the security subsystem while security policies for Web Services were being defined, which resulted in a java script error.</p> <p>Now, the method name is passed to the security subsystem in the correct format when method parameters contain array, and the java script error is no longer thrown.</p>
CR224902	Now, if a resource is deleted, its associated security policy and scoped roles are also deleted.

<b>Change Request Number</b>	<b>Description</b>
CR229823	<p>When a user with the Deployer role selected the Testing Tab, WebLogic Server threw a <code>weblogic.management.ManagementRuntimeException</code>. Thereafter, clicking on any application in the left pane would continue to generate an exception.</p> <p>Now, instead of getting the <code>weblogic.management.ManagementRuntimeException</code> in the right pane, the user gets to see the right pane with the Configuration tab selected.</p>
CR231512	<p>The Connection Policy screen for the BEA WebLogic Tuxedo Connector™ now shows the correct text.</p>
CR231600	<p>Java script pop-up windows showing cookie information are no longer displayed in any of the following scenarios:</p> <ol style="list-style-type: none"> <li>1. Log in to the Administration Console and enter the following URL:  <code>http://localhost:7001/console/actions/mbean/MBeanFramesetAction?sidebarFrameId='%5d%22%3e%3b%3cscript%3ealert%28document%2ecookie%29%3b%3c%2fscript%3e</code></li> <li>2. Open a new browser and type the following URL:  <code>http://localhost:7001/console/actions/common/SidebarAction?%3cscript%3ealert%28document%2ecookie%29%3b%3c%2fscript%3e</code></li> <li>3. Open the Administration Console (<code>http://localhost:7001/console</code>), and click the [Sign In] button with a blank password. Then the request is redirected to <code>http://localhost:7001/console/j_security_check</code>.  Then input the username as follows:  <code>weblogic"&gt;&lt;script&gt;alert (document.cookie)&lt;/script&gt;</code>  Click the [Sign In] button.</li> </ol>
CR237076	<p>On the Define Security Policy and Policy Condition pages, the Add button is no longer incorrectly displayed as the And button</p>
CR237153	<p>You can now enable Multicast Data Encryption through the Console to prevent insecure multicast messages in a clustered environment.</p>
CR238256	<p>The password is no longer incorrectly displayed as clear text in <code>server.log</code>, <code>config.xml</code>, and the Administration Console.</p>
CR239846	<p>Combo patch request of CR236810 and CR202495.</p>

## Cluster

Change Request Number	Description
CR219614	<p>When a Managed Server made an initial connection to the Administration Server using the Administration port, all subsequent connections from that Managed Server were also made using the Administration port.</p> <p>By default, connections are made to the default listen port. However, if a user has Administrator privileges, subsequent connections will be made using the Administration port.</p>
CR223302	<p>In a cluster that has the domain-wide local administration port enabled, migrating the JMS server no longer fails with a <code>NameNotFoundException</code> when WebLogic Server tries to look up an internal service called <code>weblogic.cluster.migrationControl</code>.</p>

## Connector

Change Request Number	Description
CR210249	<p>When <code>&lt;security-permission&gt;</code> is specified in the <code>ra.xml</code> file, the connector module now deploys with no problems.</p>
CR211694	<p>The WebLogic JCA handle no longer causes a memory leak. Connection handle instances are tracked through an objectID hashmap and are now cleaned up when the connection handle is closed.</p>
CR231963	<p>There is no longer a memory issue during resolution of resource references in a connector module.</p>

# CORBA

---

<b>Change Request Number</b>	<b>Description</b>
CR201403 CR213972 CR203077	<p>Previously when the client sent the close message, the server side did not close the socket because it expected more input from the client. The socket never got closed, which eventually caused connection leakage.</p> <p>Now, when the client receives a close connection message the socket is closed to prevent connection leakage.</p>
CR206054	<p>WebLogic ORB displayed a <code>NullPointerException</code> and threw a <code>CORBA.UNKNOWN</code> error instead of returning false for the <code>_is_a(String)</code> CORBA call when the string that was passed in did not match the calling CORBA object.</p> <p>Code was added to check whether the method, which is used to find the object identified by the repository identifier, returns null or not.</p>
CR220716	<p>The maximum number of file descriptors (fds) that can be specified for the <code>ioctl</code> call without causing an exception is one fewer than the operating system limit for the number of fds. This restriction is new in Solaris 10.</p> <p>To avoid getting an illegal argument exception when starting the server, the maximum number of fds is now set to one fewer than the operating system limit.</p>
CR224353	<p>A CORBA object registered with WebLogic ORB could be reached over a non-SSL port, but not over an SSL port. When the non-SSL port was disabled, an incorrect value of -1 was assigned in the IOR.</p> <p>The correct value of 0 is now assigned.</p>

---

## Core WebLogic Server

Change Request Number	Description
CR108791	<p>When the server state was <code>SHUTDOWN_IN_PROCESS</code> and <code>Runtime.getState</code> was called, WebLogic Server returned the wrong string.</p> <p>Now, WebLogic Server returns the correct string, which depicts the state of the server.</p> <p>If your applications were dependent on the wrong string constant that was being returned, you may need to change the string constant.</p>
CR128813	<p>It is now possible to create a network channel for the "admin" protocol that specifies an interface (that is, <code>ListenAddress</code>) that is different from the default network channel.</p>
CR177987 CR236405	<p>The following was an error in poll with <code>SocketException</code> that occurred with performance pack enabled:</p> <pre>java.net.SocketException: Error in poll for fd=32, revents=8</pre> <p>This problem was fixed in PerformancePack for <code>HPUX_IPF</code>.</p>
CR179278	<p>Refer to the security advisory information available at: <a href="http://dev2dev.bea.com/pub/advisory/141">http://dev2dev.bea.com/pub/advisory/141</a></p>
CR182555	<p>When JMS was migrated from server1 to server2, <code>WLIAdminMDB</code>, the Message-Driven Bean responsible for garbage collection in the WebLogic Integration Document Store, threw a <code>NameNotFoundException</code>, which caused WLI Document Store clean-up functionality to fail after migration.</p> <p>Now, Message-Driven Beans are the last components to migrate. As a result, <code>WLIAdminMDB</code> no longer throws a <code>NameNotFoundException</code> when JMS is migrated from one server to another.</p>
CR196369	<p>Refer to the security advisory information available at: <a href="http://dev2dev.bea.com/pub/advisory/138">http://dev2dev.bea.com/pub/advisory/138</a></p>
CR198659 CR208418 CR222680	<p>A corruption in the <code>TimeEventGenerator</code> data structures caused the timer thread to spin in an infinite loop due to <code>NullPointerExceptions</code>. The server had to be restarted to overcome this problem. This has now been addressed.</p>

Change Request Number	Description
CR201647 CR225246	<p>WebLogic Server threw <code>UnmarshalException</code> with nested <code>OptionalDataException</code> after EJB tier was restarted, due to synchronization issues with the <code>BasicReplicaList</code> class.</p> <p>This problem has now been fixed.</p>
CR202167	<p>When <code>beasvc</code> on Win2003 starts a server off a network share, an error no longer occurs. Now network drive mapping happens within the service itself. The following new parameters were added to the <code>beasvc</code>:</p> <ul style="list-style-type: none"> <li>• <code>localname</code></li> <li>• <code>remotename</code></li> <li>• <code>remoteuser</code></li> <li>• <code>remotepassword</code></li> </ul> <p>Example usage:</p> <pre>-localname:"X:" -remotename:"\\172.17.24.130\src810sp4" -remoteuser:"xyz" -remotepassword:"xyz"</pre> <p>You are mapping <code>\\172.17.24.130\src810sp4</code> to your local machine as <code>X:</code> and the <code>remoteuser</code>, and <code>remotepassword</code> are used to connect to the remote machine.</p>
CR202932	<p>Refer to the security advisory information available at:  <a href="http://dev2dev.bea.com/pub/advisory/153">http://dev2dev.bea.com/pub/advisory/153</a></p>
CR203320 CR220927	<p>A deadlock no longer occurs between Multicast send and receive calls.</p>
CR203523 CR228079	<p>The following log message has been reinstated to appear as it had in previous WebLogic Server releases.</p> <pre>&lt;Warning&gt; &lt;WebLogicServer&gt; &lt;000333&gt; &lt;Queue usage is greater than QueueLengthThresholdPercent "3%" of the maximum queue size. We'll try to allocate ThreadsIncrease "5" thread(s) to help.&gt;</pre>

Change Request Number	Description
CR208567	<p>A Managed Server referenced a listener thread that listened for the commands or responses sent by the Managed Server. When a Node Manager had gone down, the Managed Server tried to re-initiate the connection to the Node Manager. After re-establishing the connection, the Managed Server created the new listener thread and started it, but did not change the reference to the listener. It still pointed to the old listener.</p> <p>Now, the Managed Server sets the reference to a listener properly after re-establishing a connection to the Node Manager.</p>
CR209153	<p>When remote-client-timeout in EJB descriptors was set to a value equal to or greater than 33 seconds, the runtime descriptor did not generate properly. Instead, remote-client-timeout was ignored.</p> <p>Now, a value greater than 32 seconds can be specified for the remote-client-timeout setting in the EJB descriptors.</p>
CR209595 CR234506	<p>The constant pool index overshoot the <code>Byte.MAX_VALUE</code> (<math>2^7 - 1</math>) in the bytecode generation code, which caused EJB deployment to fail with a <code>java.lang.VerifyError</code> stating the underlying cause as <code>Illegal type in constant pool</code>.</p> <p>To fix the problem with the bytecode generation, the index to <code>Short.MAX_VALUE</code> (<math>2^{15} - 1</math>) has been extended.</p>
CR210128 CR210838 CR214575	<p>The socket layer no longer throws a <code>NullPointerException</code> when an RMI layer throws a <code>QueueFullException</code>.</p>
CR210920 CR233948	<p>A prepared statement that was cached with a pooled connection, which failed with an exception, is now completely removed from the cache, and no longer results in a memory leak.</p>
CR211299	<p>Refer to the security advisory information available at:  <a href="http://dev2dev.bea.com/pub/advisory/156">http://dev2dev.bea.com/pub/advisory/156</a></p>

Change Request Number	Description
CR212744 CR221785	The <code>SSOLIOContextTable.registerForThrottling()</code> gets the reference to the <code>ServerThrottle</code> instance inside a synchronized region. If the <code>ServerThrottle</code> had not been instantiated, the constructor was called, which sometimes resulted in deadlock because the <code>ServerThrottle</code> was instantiated holding the <code>SSOLIOContextTable.class</code> lock. To avoid the deadlock, WebLogic Server now gets the reference to the <code>ServerThrottle</code> instance before obtaining the <code>SSOLIOContextTable.class</code> lock.
CR213915	When the remote server to which the stub is connected is restarted, the stub now fails over to the other servers in the cluster.
CR215524	Refer to the security advisory information available at: <a href="http://dev2dev.bea.com/pub/advisory/157">http://dev2dev.bea.com/pub/advisory/157</a>
CR217262	If <code>QueueLength</code> reached maximum, it resulted in a <code>FullQueueException</code> , which in turn resulted in server shutdown. To fix this, WebLogic Server now handles the <code>FullQueueException</code> and also closes the socket for SSL before the handshake so that the clients get disconnected properly.
CR217423 CR241142	WebLogic Server can now accept <code>POST/GET</code> requests from a client after the <code>KeepAlive</code> connection is timed out on Linux.
CR217854 CR221776	When WebLogic Server threw a <code>SecurityException</code> , as a result of a remote call made to the Administration port by an anonymous user, it failed to return the transaction context with the response. As a result, the client was unable to decrement the request count resulting in a <code>CommitException</code> . Now, when WebLogic Server throws a <code>SecurityException</code> , it also sends the transaction context.
CR220409	Combo patch request of CR217423 and CR201403.
CR222932	When standalone JMS consumers listening on queues in WebLogic Server were created and the client was killed abruptly, consumers on the server side were still alive. Restarting the client created additional consumers with the same stub and then, if a <code>peerGone</code> exception was thrown, only one listener received the message. One stub can now have two listeners, which resolves the problem.
CR224570	A synchronization problem caused two different requests to be assigned the same request ID. This caused one of the requests to hang. Now a unique ID is assigned to each request.

## Resolved Problems for Service Pack 5

<b>Change Request Number</b>	<b>Description</b>
CR229267	The forced undeployment of applications after graceful shutdown of WebLogic Server timeout sometimes lead to <code>ConcurrentModificationException</code> . WebLogic Server no longer throws this exception while undeploying applications.
CR232983	WebLogic Server no longer throws a <code>peerGone</code> exception when the <code>ConnectionManager</code> was waiting to acquire a lock to send heartbeats. This problem has been fixed.
CR236961	WebLogic Server now propagates non-public interfaces.
CR239324	If a socket was unregistered and then registered, and if between the two calls the timeout timer never ran, the data structures were getting out of sync, which prevented the newly registered socket from ever timing out. The new register for the socket is no longer ignored by a previous unregister.

# Deployment

---

<b>Change Request Number</b>	<b>Description</b>
CR190015	When tasks were purged from the Administration Console, a
CR220929	<code>weblogic.utils.AssertionError</code> was thrown. Now, while completed tasks inside the deployment code are deleted, additional checks in the deployment code determine whether the MBean is still registered with the MBean Server or whether it has already been unregistered.
CR220155	In clustered deployments with <code>enforceClusterConstraints</code> set, the application deployment fails if one or more servers in the cluster is not up. Changes were made so that when these servers come up later, they continue using the application bits available with them and do not deploy the new application bits. Now, the same application version runs on all the servers of the cluster even after deployment failure.
CR220398	A <code>SocketException</code> is no longer thrown when you deploy a large application (over 100M).
CR236855	When deployments are pinned to clusters, the <code>ApplicationLifecycle</code> listener is no longer invoked on non-targeted servers of the cluster.

---

# EJB

---

<b>Change Request Number</b>	<b>Description</b>
CR112185	Now, ejbc logs a warning message if there is a mismatch in the <code>jndiName(s)</code> and type of view(s) defined by an EJB. For example, if the bean, <code>TraderBean</code> , defines a local view and has no <code>local-jndi-name</code> defined in the <code>weblogic-ejb-jar.xml</code> file, ejbc reports the following warning: The ejb 'TraderBean' has a Local View. However, the <code>&lt;local-jndi-name&gt;</code> is not set.

---

Change Request Number	Description
<p>CR125400 CR132496</p>	<p>A startup class that calls one EJB business method on doSomethingOnA of BeanA. This method gets the home interface for BeanB and this leads to the following ClassCastException:</p> <pre>&lt;29 janv. 03 13:21:19 CET&gt; &lt;Info&gt; &lt;EJB&gt; &lt;010051&gt; &lt;EJB Exception during invocation from home: test_ccex.a.BeanABean_124zg1_HomeImpl@5c5ca2 threw exception: java.lang.ClassCastException: Cannot narrow remote object to test_ccex.b.BeanBHome java.lang.ClassCastException: Cannot narrow remote object to est_ccex.b.BeanBHome at weblogic.iiop.PortableRemoteObjectDelegateImpl.narrow(Por tableRemoteObjectDelegateImpl.java:223) at javax.rmi.PortableRemoteObject.narrow(PortableRemoteObjec t.java:132) at test_ccex.a.BeanABean.doSomethingOnA(BeanABean.java:49) at test_ccex.a.BeanABean_124zg1_EOImpl.doSomethingOnA(BeanAB ean_124zg1_EOImpl.java:46) at java.lang.reflect.Method.invoke(Native Method) at com.csg.am.dcf.weblogic.startup.BeanExecutor.startup(Bean Executor.java:183) at weblogic.t3.srvr.StartupClassRunner.invokeStartup(Startup ClassRunner.java:141) at weblogic.t3.srvr.StartupClassRunner.invokeClass(StartupCl assRunner.java:122) at weblogic.t3.srvr.StartupClassRunner.access\$0(StartupClass Runner.java:113) at weblogic.t3.srvr.StartupClassRunner\$1.run(StartupClassRun ner.java:86) at weblogic.security.service.SecurityServiceManager.runAs(Se curityServiceManager.java:744) at weblogic.t3.srvr.StartupClassRunner.run(StartupClassRunne r.java:71) at java.lang.Thread.run(Thread.java:479)</pre> <p>A code fix resolved a classloading issue in RemoteObjectReplacer.</p>
<p>CR126351 CR132693 CR133769</p>	<p>The appc generated compiler errors if the value of the &lt;table-name&gt; element in the weblogic-cmp-rdbms-jar.xml DD contained special characters such as '.' and '@'.</p> <p>The DB2 database now allows table names that include these special characters. Illegal Java characters are now replaced by '_' in the generated code. This change is also applicable to CMP beans that are mapped to multiple tables.</p>

<b>Change Request Number</b>	<b>Description</b>
CR132510	When enabling the relationship caching on an entity bean that is using optimistic concurrency with <code>cache-between-transaction</code> set to true, WebLogic Server no longer throws an <code>IllegalStateException</code> .
CR133421	In a domain with a JMS Server that has no targets, a Message-Driven EJB deployment with distributed destination or migratable destination no longer fails with an <code>ArrayIndexOutOfBoundsException</code> .
CR174593 CR198357	WebLogic Server used to throw a <code>ConcurrentModificationException</code> when: <ul style="list-style-type: none"> <li>• the finder of an entity bean was invoked in the <code>ejbStore</code> method</li> <li>• new entity beans were enrolled in the transaction in the <code>ejbStore</code> method</li> <li>• modifying entity beans of the same type in the <code>ejbStore</code> method</li> </ul> This exception is no longer thrown.
CR182398	EJBC no longer fails if a CMP Entity Bean has a CMP field named <code>oldState</code> .
CR186266	Setting the Extra EJB Compiler Options to the value <code>-J-mx512M</code> in the Administration Console and then deploying an EAR file no longer causes deployment failure and the following exception: <pre>Compiler failed executable.exec</pre> Additionally, EJB compilation is now a forked process if there are any extra <code>ejbc</code> options or if any compiler other than the default is specified. WebLogic Server now supports Java heap size options in the following formats as part of the extra <code>ejbc</code> options: <pre>-J-mx128m</pre> <pre>-J-Xmx128m</pre>
CR186762	Now, even when a transaction only modifies blob or clob fields in a bean, the version or timestamp column in a table is updated.
CR194306 CR211196	Using “select for updates” with optimistic concurrency EJBs led to performance issues. The “select for updates” operation is no longer used for optimistic concurrency EJBs that use Oracle drivers that implement the proprietary <code>sendBatch()</code> API. As a result, for optimistic failures, it is no longer possible to report errors at the pk level.

<b>Change Request Number</b>	<b>Description</b>
CR197625	<p>When a remove call for a CMP bean failed because of a database constraint violation, WebLogic Server failed to destroy the bean at commit time. This caused the bean to be reused without being initialized, resulting in <code>NoSuchObjectException</code> when the bean was subsequently accessed.</p> <p>The bean is now destroyed in the cache.</p>
CR200204	<p>Now, when an EJB that has both local and remote interfaces is deployed and being accessed from a Web client deployed in the same application (EAR), a <code>ClassCastException</code> is no longer thrown while create is being called on <code>LocalHome</code>.</p>
CR202817 CR237811	<p>In a cluster, when a child bean is added to the parent bean on one server; the beans use the Optimistic Concurrency strategy; and cache-between-transactions is set to true, the CMP/CMR field no longer returns a stale set of child beans.</p>
CR203644	<p>Replicated Stateful Session Beans in a cluster with InMemory replicated session no longer throw a <code>NoSuchObjectException</code> or a <code>LeasedRemoteRef</code> error after instances are passivated or after the cluster instance is shut down.</p>
CR205974	<p>When a collection valued CMR field is accessed in a transaction other than the one in which it is created, WebLogic Server no longer throws an <code>IllegalStateException</code>.</p> <p>As a result, WebLogic Server delivers the correct result set from the finder query regardless of whether caching is turned on or off.</p>
CR206209	<p>BEA implemented optimizations for immutable fields of CMP EJBs such that the <code>setXXX()</code> method does not write the values of unchanged primitive and immutable fields to the database. This optimization improves performance, especially in applications with a high volume of database updates.</p> <p><code>BigInteger</code> and <code>BigDecimal</code>, which are immutable classes, were excluded from this optimization and have now been included.</p>
CR208540	<p>WebLogic Server no longer queries against the database for <code>findByPK</code> calls after a bean is reloaded, but instead retrieves the information from the cache.</p>
CR209678	<p>When the <code>ejbCreate()</code> method was declared as <code>final</code>, <code>ejbc</code> correctly flagged an error, but the error message content was incorrect. The error message was corrected as follows:</p> <p>the <code>ejbCreate</code> method(s) must not be declared as <code>final</code></p>

Change Request Number	Description
CR209908 CR210116	<p>When an <code>ejbSelect</code> query returned a <code>ResultSet</code> (WebLogic Server-specific), the JDBC objects were not released after the query was done, which resulted in a connection leak.</p> <p>JDBC objects are now released in the finally block of the method. As a result, there is no longer a JDBC connection leak.</p>
CR209987 CR210650 CR229780	<p>Assuming <math>n</math> as the max cache size specified in the deployment descriptor, when trying to execute an EJB finder call returning more than <math>n</math> records, WebLogic Server throws a <code>CacheFullException</code>. This is expected because the Entity EJB cache size is set to <math>n</math>. However, subsequent finder calls on the same EJB that returns a single record failed with the <code>CacheFullException</code> because the cache size had already been incremented, and it was not decremented when the <code>CacheFullException</code> occurred.</p> <p>Now, subsequent finder calls on the same EJB no longer fail with this exception because the cache size is decremented whenever the <code>CacheFullException</code> occurs.</p>
CR210673	<p>If the data type of a <code>cmp</code> field was <code>byte[]</code>, <code>ejbSelects</code> returning collection types using that field were not generated properly.</p> <p>For example, for an <code>ejbSelect</code> defined as:</p> <pre>SELECT o.identifier from TestByteArray as o where the cmp-field "identifier" maps to byte[] in the bean class, ejbc created an unknown return data type in the generated class and compilation failed:</pre> <pre>( [B __WL_retVal = null; )</pre> <p>Now, the java source type is generated correctly:</p> <pre>byte[] __WL_retVal = null;</pre>
CR210675	<p>When a <code>CMP</code> field was defined as type <code>String</code>; <code>SybaseBinary</code> in the deployment descriptor, <code>ejbc</code> generated classes were still using <code>setBinaryStream()</code> instead of <code>setBytes()</code> in case of finders.</p> <p>Now, the RDBMS generated code always uses <code>setBytes()</code> rather than <code>setBinaryStream()</code> for finder queries if the query parameter is of type <code>byte[]</code> and the <code>CMP</code> bean contains at least one <code>CMP</code> field that defines <code>dbms-column-type</code> as <code>SybaseBinary</code>.</p>
CR210903 CR224454	<p>WebLogic Server no longer throws a <code>ClassCastException</code>, as it does not try to cast a <code>BMP</code> bean into a <code>CMPBean</code> in the <code>afterCompletion</code> callback.</p>

Change Request Number	Description
CR212483	<p>When a finder was executed within the <code>ejbRemove()</code> body with <code>include-updates</code> set to true, the bean on which <code>ejbRemove</code> was called was deleted before the <code>ejbRemove</code> call was completed in the <code>flushModifiedBeans</code> operation. This happened when <code>order-database-operations</code> was set to true.</p> <p>The bean is now deleted after the <code>ejbRemove</code> call is completed.</p>
CR215848	<p>Serializing access to create the execute queue and then start threads for that queue, ensures that when Message Driven Beans are deployed and WebLogic Server uses asynchronous polling, a thread - <code>'&amp;quot;weblogic.ejb20.internal.JMSMessagePoller&amp;quot;'</code> is created for every Message Driven Bean deployed. Based on the load, child pollers are created to handle the load as and when required.</p> <p>Synchronization changes have been implemented to ensure that a race condition does not occur when Message Driven Beans fail to connect at deployment time. Note that while this problem has been resolved, performance might be affected slightly.</p>
CR217172	<p>In the same transaction, accessing a <code>ReadOnly</code> bean with <code>include-updates</code> set to <code>False</code> caused pending changes to be flushed to the database when the finder parameters were of type primitive array such as <code>byte[]</code>. This situation degraded performance, caused a lock on the database side, and caused all other transactions to be held up until the transaction performed a commit or rollback.</p> <p>Now, <code>include-updates</code> set to <code>False</code> for any finder query work correctly for any type of method parameters, and changes that should not be flushed to the database are no longer flushed.</p>
CR217188	<p>BEA-012034 was added to the list of warnings that can be disabled with the <code>&lt;disable-warnings&gt;</code> element.</p>
CR217605	<p>If an EJB has a run-as security role configured, WebLogic Server no longer uses different security principals while executing different methods on the EJB class. WebLogic Server now sets the correct run-as principal for all the EJB methods.</p>
CR217859	<p>The correct transaction timeout value is now used when a transaction is created in <code>JMSMessagePoller/ContinuousJMSMessagePoller</code>.</p>
CR219935	<p>When security providers are implemented with EJBs, local stack corruption, which resulted in a stateless session bean leak, no longer occurs. WebLogic Server maintains the stack order correctly.</p>

<b>Change Request Number</b>	<b>Description</b>
CR220559 CR235671	When the commit setting for the delay-database-insert-until flag in WebLogic Server 7.0 style descriptors is used, the WebLogic container now correctly enables both batch operations and order database operations.
CR220607	The bean on which the <code>ejbRemove()</code> method is called is now available within the scope of its own <code>ejbRemove()</code> method when <code>order-database-operations</code> is set to <code>True</code> .
CR220665	<p>Even if the <code>CMP</code> field was byte array, it was reserialized again to byte array while being put into the database, which resulted in corruption of blob data.</p> <p>Now, WebLogic Server no longer reserializes a byte array to get another byte array unless <code>serialize-byte-array-to-oracle-blob</code> is explicitly set to <code>True</code>. As a result of this fix, blob data is no longer corrupted.</p>
CR222244	<p>In WebLogic Server 8.1 Service Pack 3, CR128063 was fixed as follows:</p> <p>When an MDB suspended a transaction, if the transaction was later resumed, it was committed or rolled back using a different thread. This caused problems for some XA implementations.</p> <p>The MDB container now ensures that the transaction is started and committed using the same thread.</p> <p>In WebLogic Server 8.1 Service Pack 5, CR184585 was fixed as follows:</p> <p>Added support for <code>XAResource</code> implementations that possess thread affinity behaviors.</p> <p>As a result of the fix for CR184585, the fix for CR128063 was reverted in WebLogic Server 8.1 Service Pack 5.</p>
CR223475	<p>The Inter-domain trust is no longer needed for a Message-Driven Bean with the <code>"NotSupported"</code> transaction attribute. Also the Credential Mapper needs to be configured to map the remote credentials to run-as-user in a Message-Driven Bean domain.</p> <p>In addition, JMS now uses consumer credentials when pushing exceptions to the consumer.</p>
CR227533 CR217872 CR239336	Stateless beans now reflect the correct <code>EJBTransactionRuntime</code> attributes.
CR228320	Object array type parameters used in Finder methods are no longer ignored; they are now parsed by WebLogic Server.

## Resolved Problems for Service Pack 5

<b>Change Request Number</b>	<b>Description</b>
CR228579 CR234062	The entity beans that return true for an <code>isIdentical()</code> call now also return true for an <code>equals()</code> call.
CR230446	The CMP EJB beans marked as "isRemoved" are no longer returned as the result of an internal (relationships)/external(user defined) finder call.
CR232022	A Message-Driven Bean could not connect to a Foreign JMS Server that had security enabled. An MDB can now connect to a Foreign JMS Server when the Credential Mapper or Foreign JMS wrapper is configured properly.
CR235103	During JMS migration, a <code>NullPointerException</code> is no longer thrown during migration of an undeployed MDB.
CR235944	While doing a compliance check, the <code>EJBObjectComplianceChecker</code> no longer ignores a few business methods.
CR237440	Using <code>MetaData</code> for to validate DB schema no longer leads to a mismatch between column names and corresponding column types.
CR240444	Combo patch request of CR188814 & CR209908.

## Installer

---

<b>Change Request Number</b>	<b>Description</b>
CR195440	Node Manager installed as an NT service with the JVM set to JRockit no longer has a problem starting the service.

---

## J2EE

---

<b>Change Request Number</b>	<b>Description</b>
CR178426 CR211255	The servlet <code>destroy()</code> method is now called during the deactivate phase of two-phase undeployment.  As a result, Web Application sessions are removed in the deactivate phase instead of in the rollback phase. This behavior is modified so that the servlets can access the EJB or app-scoped JDBC pools from the servlet <code>destroy()</code> method.
CR222757	Compilation errors in a JSP Web application caused <code>appc</code> to stop compiling the JSP.  A new option, "-k", has been introduced in <code>appc</code> to continue JSP compilation even after a compilation error. For <code>wlappc</code> tasks, a new attribute "continueCompilation" has been added.

---

## jCOM

---

<b>Change Request Number</b>	<b>Description</b>
CR208772	The J-Integra version bundled with WebLogic Server 8.1 is upgraded to J-Integra 2.3.

---

## JDBC

Change Request Number	Description
CR087241 CR216714	<p>A JDBC application using XA calls and any Oracle driver was getting an ORA-ORA-01002: fetch out of sequence error while fetching rows if a transaction was suspended and resumed.</p> <p>The <code>AddOracleFlagToXAResource</code> flag has been added to fix this problem when the XA calls use the Oracle 10g thin driver. To avoid getting the ORA-ORA-01002: fetch out of sequence error while using XA calls, you must use the Oracle 10g thin driver and turn on the <code>JDBCConnectionPool</code> flag.</p>
CR136497	Calling the JTS connection <code>getWarnings()</code> method after a transaction is finished no longer results in a transaction over exception.
CR136713	The Administration Console now properly displays leaked connections for the JDBC connection pool.
CR136713 CR226682	Leaked connections for a JDBC connection pool are now properly displayed in the Administration Console when you click "View Leaked Connections.." in the pop-up menu for the JDBC connection pool.
CR17325	<p>The Oracle thin driver only allows its property, <code>SetBigStringTryClob</code>, to be set through a non-JDBC-standard <code>DataSource</code> method. Because WebLogic Server only uses standard methods to set properties, <code>SetBigStringTryClob</code> was not sent successfully from the pool to the Oracle 10g driver.</p> <p>Now, WebLogic Server determines whether the <code>DataSource</code> is an Oracle thin XA <code>DataSource</code>, and calls the nonstandard method to set the pool's driver properties. Thus, if the pool includes <code>SetBigStringTryClob=true</code>, this property, and any other Oracle-specific properties that need to be set that way, will be set.</p>
CR189649	<p>When the Sybase database is used with Container Managed Beans and the Sybase driver, and when both the method that contains transaction as <code>Required</code> and the method that contains transaction as <code>NotSupported</code> are called up in sequence from the client side, the following exception no longer appears in the JDBC Log file of the server:</p> <pre>com.sybase.jdbc2.jdbc.SybSQLException: SET CHAINED command not allowed within multi-statement transaction.</pre>

Change Request Number	Description
CR189956	<p>Using <code>ADMIN_JNDI_NAME</code> with <code>JDBCTxDataSource</code> (using a JTS Driver) when performing a lookup on the Administration Server resulted in a <code>javax.ejb.TransactionRolledbackLocalException</code> being thrown if the Administration Server was down.</p> <p>Now, for a JTS Driver, WebLogic Server does a <code>LOCAL_JNDI_NAME</code> lookup first and then does the <code>ADMIN_LOOKUP</code> only if <code>datasource</code> is not available locally. As a result, no exception is thrown if the lookup is made by calls within the Managed Server for <code>JDBCTxDataSource</code>.</p>
CR195721	<p>Previously, you could not use a semicolon as part of the JNDI DataSource name because the semicolon is the default character used to bind multiple JNDI names.</p> <p>The semicolon is still the default character that binds multiple JNDI names. However, now you can use the <code>JNDINameSeparator</code> flag to define the character that binds JNDI names for <code>JDBCDataSource</code> and <code>JDBCTxDataSource</code>. If you choose a character other than a semicolon, you can then use the semicolon as part of the JNDI DataSource name.</p>
CR197163 CR215627	<p>The JDBC connection pool's testing of connections was consuming a lot of database resources because each test was creating a new plain statement which requires the DBMS to parse and plan the test SQL every time.</p> <p>Now, pools will reuse a single prepared statement for a connection's test which results in improved performance. However, if any application DBMS tables or procedures are referred to in the test SQL and if they structurally change at runtime, such as an index being added, this may invalidate the test <code>PreparedStatement</code>'s query plan. As a result, the subsequent test will fail and the connection and test statement will be replaced. The test SQL suggested by the Administration Console will typically not include any structurally changing table, so the problem of needlessly recycling a connection is now minimized.</p>
CR199344 CR220931	<p>When an anonymous user attempts to retrieve DataSource MBean, WebLogic Server no longer throws the following error message:</p> <pre data-bbox="357 1298 1193 1350">java.lang.SecurityException: User &lt;anonymous&gt; does not have access to the administrator port</pre>
CR199447	<p>Now, when an <code>XAER_RMERR</code> exception is thrown, the JDBC Connection is returned to the pool.</p>
CR202067	<p>If <code>CountOfRefreshFailuresTillDisable</code> is set for the pool, it will disable itself if it cannot make new connections. Now the pool will correctly, periodically try to reconnect and will re-enable itself when it is able to reconnect.</p>

Change Request Number	Description
CR203460	<p>When XAConnectionFactory established a pool connection, it ignored the value of the seconds-to-trust property and instead used the default value.</p> <p>This problem has been resolved.</p>
CR205140	<p>In WebLogic Server versions prior to 8.1, all pool self-tests were suspended when the pool was suspended. WebLogic Server 9.0 reinstates this functionality.</p>
CR205179 CR205771	<p>JDBC DataSource now recognizes the Oracle RAC xid propagation delay and, during an RAC failure, reissues commit and rollback after the first commit and rollback fails.</p>
CR205768	<p>Oracle RAC exhibited a delay when replicating the state of pending (prepared) Xids from one RAC instance to another. This delay caused mixed transaction outcomes when the WebLogic Server Transaction Manager attempted to resolve global transactions that contained RAC participants.</p> <p>Enhancements to the WebLogic Server Transaction Manager recovery service have resolved this problem.</p>
CR206759	<p>WebLogic Server now uses <code>ShrinkFrequencySeconds</code> instead of <code>shrinkPeriod</code> when setting attributes on <code>JDBCConnectionPoolMBean</code> through an Ant task. As a result, JDBC <code>shrinkPeriodMins</code> now sets Shrink Frequency in the Administration Console.</p>
CR207293	<p>When a WebLogic Server 8.1 instance performs a remote RMI JDBC connection to a WebLogic Server 6.1 instance, the connection no longer closes prematurely.</p>
CR208550	<p>The <code>wlconfig</code> Ant Task in WebLogic Server 8.1 Service Pack 2 and earlier produces an error message if a database connection pool is not configured properly. However, WebLogic Server 8.1 Service Packs 3 and 4 do not show an error if a database connection pool is not configured properly.</p> <p>The error message is displayed properly in WebLogic Server 8.1 Service Pack 5.</p>
CR208691 CR212597	<p>The monitoring page for the JDBC pool in the Administration Console no longer indicates the presence of waiters when none exist.</p>
CR209007	<p>The method <code>ResultSet.wasNull()</code> now returns the correct information when the Weblogic Datasource is used.</p>
CR210219	<p>When WebLogic Server JDBC pools run their periodic test of idle connections, they reserve, test, and release one connection at a time. Temporary outages (<code>ResourceExceptions</code>) no longer occur when a new user load comes in.</p>

<b>Change Request Number</b>	<b>Description</b>
CR210818	Pool resources are no longer reclaimed during shutdown. During WebLogic Server shutdown, WebLogic Server no longer makes <code>getConnection</code> calls while connections are being destroyed.
CR210859	<code>Statement.getUpdateCount</code> no longer returns -1 when a Sybase type 4 driver is used in a WebLogic Server pool.
CR211464	When <code>InactiveConnectionTimeout</code> is configured to an amount of time, truly inactive connections are now returned to the connection pool, and connections waiting for DBMS response are not returned.
CR211662	WebLogic Server provides the interface for Oracle thin driver Clobs and Blobs to give you access to certain Oracle-specific methods. This interface is no longer missing the <code>trim()</code> method and the <code>truncate()</code> method.
CR211812	The transaction coordinator was racing with the application to close or cancel a statement, which resulted in a <code>java.lang.NullPointerException</code> .  To avoid this race condition, WebLogic Server now ensures that the connection is viable before calling the underlying code. As a result, a <code>java.lang.NullPointerException</code> is no longer thrown when a statement is closed or cancelled.
CR212051 CR221756	The <code>getAutoCommit()</code> method always returned <code>False</code> , whether the connection was inside or outside a distributed transaction.  Now, <code>getAutoCommit()</code> returns <code>True</code> when the <code>XAConnection</code> object is outside a distributed transaction. Pooled connections present as the JDBC 3.0 specification mandates: <code>autoCommit(true)</code> outside a global transaction.  If you are using a Sybase driver, you must download and install the Sybase driver version that is later than the version shipped with WebLogic Server 8.1 Service Pack 4.
CR212779 CR205768	The JDBC configuration attributes, <code>XARetryDurationSeconds</code> and <code>XARetryIntervalSeconds</code> , were added to the <code>JDBCConnectionPoolMBean</code> . These attributes were also added to the Administration Console so that you can set them without manually editing the <code>config.xml</code> file.
CR213774 CR230719	<code>WLCachedRowSet</code> now returns the time portion (hour, minute, and second) from an Oracle database date column.

<b>Change Request Number</b>	<b>Description</b>
CR214267	JTA JDBC connections now keep track of statements so they can be cancelled at transaction timeout.
CR215076 CR222549	Now, even if <code>CountOfRefreshFailuresTillDisable</code> is greater than 1, the pool will disable itself after that many consecutive failures to refresh, as documented.
CR216078	Combo patch request of CR212674, CR186754, CR188329, and CR172353.
CR219411	Connection pool refresh is supposed to test all unused connections. It was testing one idle connection <i>n</i> number of times. Now it properly tests <i>n</i> number of idle connections one time each.
CR219822 CR212898	In the Examples domain that shipped with WebLogic Platform 8.1 Service Pack 3, if the thread ID that was logged as part of the statement profiling contained XML reserved characters, many XML errors occurred when <code>getStatementProfiles()</code> was called on the <code>JDBCConnectionPoolRuntime</code> MBean.  WebLogic Server now ensures that the pool profiling information no longer contains any XML reserved characters.
CR221966	WebLogic Server no longer makes more than the necessary number of transaction rollbacks in the Oracle database.
CR222454	The <code>weblogic.jdbc.wrapper.XAConnection.resetTransactionIsolation()</code> method no longer leaks transactions.
CR225187	The driver type was incorrect for the SQL/MX driver in the <code>jdbcdrivers.xml</code> file.  The driver entry for the SQL/MX driver was removed from the <code>jdbcdrivers.xml</code> file. As a result, you now must manually enter all driver and property information while defining a pool to use the SQL/MX driver.
CR227920	In some cases, redeploying an application on a Managed Server from the Administration Console, after having restarted the Administration Server, threw a <code>java.rmi.ConnectException</code> .  WebLogic Server no longer throws this exception.

<b>Change Request Number</b>	<b>Description</b>
CR228597	<p>Now, <code>CallableStatement</code> contains the same processing that <code>PreparedStatement</code> contains. When you use <code>CallableStatement</code>, WebLogic Server supplies proxy processors for certain non-serializable arguments to JDBC calls such as Input streams.</p> <p>As a result, using <code>CallableStatement</code> to call a stored procedure in the 10g database to insert a CLOB field no longer results in a <code>java.rmi.MarshalException</code>.</p>
CR229318	<p>While processing JMS messages, if the transaction times out, the database rows are no longer locked.</p>
CR229477	<p>If you reset the connection pool with <code>TestFrequencySeconds</code> using the Administration Console, WebLogic Server no longer throws the following exception:</p> <pre>java.sql.SQLException: Statement is closed</pre>
CR233641	<p>The <code>ResultSetRowCache</code> class does not obtain the complete time stamp from the DATE column of an Oracle database.</p> <p>To resolve the problem, contact BEA Customer Support for the latest patch.</p>
CR233921	<p>When closing an RMI JDBC statement, Weblogic Server no longer fails to close the result sets.</p>
CR236171	<p>Differences between the WebLogic Server 7.0 and 8.1 releases in the implementation of the <code>JDBCConnectionPoolRuntimeMBean</code> no longer result in migration problems.</p>
CR236988	<p>Lack of synchronization caused the <code>JTACConnection</code> object to throw a <code>NullPointerException</code>.</p> <p>This problem has been fixed.</p>
CR243411	<p>A JDBC <code>NullPointerException</code> was thrown during a <code>MultiPool</code> failover using Oracle 10g RAC.</p> <p>This problem has been resolved. Contact BEA Customer Support for the patch.</p>

## jDriver

---

<b>Change Request Number</b>	<b>Description</b>
CR197516	The ExecuteBatch method in the Statement class no longer always returns an array of 0 length when used in the jDriver for Oracle.
CR228794	Increasing the statement's fetch size after using it no longer causes native memory problems that can cause the JVM to crash.

---

## JMS

---

<b>Change Request Number</b>	<b>Description</b>
CR177558	A server-side memory leak no longer occurs with NO_ACKNOWLEDGE delivery mode.
CR187610	<p>JMS Bridge was not using the Messaging Bridge Thread Pool for bridge dispatch requests. Now, JMS Bridge uses the Messaging Bridge Thread Pool if one has been configured. If there are not enough available threads configured in the Messaging Bridge Thread Pool, the bridge uses the default execute thread pool.</p> <p>Bridge also logs a warning message if the configured Messaging Bridge Thread Pool size is insufficient and then attempts to obtain a thread from the default execute pool instead.</p>

---

Change Request Number	Description
CR190098	<p>Now, <code>session.recover()</code> can be called for single instance non-transacted topic Message-Driven Beans. As a result, messages get re-delivered when they use JMS Topic with a Message-Driven Bean listening on it.</p> <p>WebLogic Server does not allow <code>session.recover()</code> on multi-instance non-transacted topic Message-Driven Beans. If you want to use non-transacted topic Message-Driven Beans and would like to have <code>session.recover()</code> called on <code>onMessage()</code> you must use a single instance of the topic Message-Driven Bean as in the below example code:</p> <pre data-bbox="357 652 1159 777">&lt;pool&gt; &lt;max-beans-in-free-pool&gt;1&lt;/max-beans-in-free-pool&gt; &lt;initial-beans-in-free-pool&gt;1&lt;/initial-beans-in-free-pool&gt; &lt;/pool&gt;</pre> <p>If you need multi-instance topic Message-Driven Beans for performance reasons and also need recovery you should use transacted Message-Driven Beans.</p> <p>Non-transactional single instance topic Message-Driven Beans that throw an exception in <code>onMessage()</code> get a re-delivery on the message.</p>
CR200474	<p>Java level deadlock no longer occurs with topic subscribers using multicast on subscriber <code>JMSConsumer.close()</code>.</p>
CR200503 CR210126	<p>WebLogic Server incorrectly reported a thread as a stuck thread when that thread was being used by JMS to deliver multiple messages when no new requests were on the <code>ExecuteQueue</code>. WebLogic Server now checks for this condition so as not to report a false positive.</p>
CR202306	<p>You can now use the Administration Console to clear the multicast address field of a JMS Topic.</p>
CR202905 CR206889 CR230513	<p>Setting the local JNDI name on a newly-created Foreign JMS Connection Factory using the WebLogic Server Scripting Tool (WLST) no longer results in any <code>NullPointerException</code>s.</p>
CR203841 CR221782	<p>WebLogic Server no longer throws the informational exception when JMS is unable to find a distributed destination member in the JNDI table.</p>
CR204981	<p>To prevent forwarding of messages with null message bodies, JMS has been modified to ensure that it pages in any necessary message bodies before forwarding the messages to distributed topic members</p>

Change Request Number	Description
CR206346	<p>JMS was load balancing twice in an attempt to locate a member that met the message criteria when the following circumstances were all true:</p> <ol style="list-style-type: none"> <li>1. the sender was created on a null destination, and</li> <li>2. the message delivery mode required persistence, and</li> <li>3. the first load balance attempt (made to resolve null destination) did not locate a member with a persistent store configured.</li> </ol> <p>Double load balancing attempts no longer occur in the preceding circumstances.</p>
CR206820 CR221783	<p>Producers now load-balance correctly to distributed queues without encountering security exceptions on some members when a domain wide administration port has been enabled.</p>
CR209263	<p>MQSeries ran out of channels because WebLogic Server had too many sessions open. Now, WebLogic Server closes both sessions and XAsessions, which allows MQSeries to close associated channels and prevents MQSeries from running out of channels.</p>
CR210888	<p>Redelivery delay no longer causes an unexpected stall in message delivery.</p>
CR211569	<p>WebLogic Server no longer throws an <code>IllegalStateException</code> when committing a transacted JMS session when using a JMS Wrapper in a Servlet.</p>
CR214523 CR235097	<p>To avoid hanging threads, a <code>connection.stop()</code> method now allows the WebLogic Messaging Bridge source queue to drain whatever is in the pipeline before stopping the connection.</p>
CR215637	<p>JMS no longer throws an unexpected exception when attempting to delete a duplicate durable subscriber on JMSServer boot.</p>
CR215767	<p>The following configurable WebLogic Server command line system property was added to override the current default Max Session Pool setting:</p> <pre>-Dweblogic.deployment.jms.MaxSessionPoolSize=10000</pre>
CR216063 CR234758	<p><code>MessagingBridgeRuntime</code> mbean <code>getState()</code> is now properly refreshing state information.</p>

<b>Change Request Number</b>	<b>Description</b>
CR216945	<p>The WebLogic Messaging Bridge failed to create source consumers and message listeners as a result of a <code>JMSecurityException</code> that occurred when the JMS source destination had a security role defined that prevented <code>user=anonymous</code> from receiving on the source destination.</p> <p>The WebLogic Messaging Bridge was modified to create a consumer using JNDI initial context credentials for the source destination.</p>
CR217908	<p><code>JMSConnectionRuntimeMBean.getHostAddress()</code> now returns the proper IP address of the client JVM in the proper format.</p>
CR219243	<p>When attempting to create <code>JMSQueue</code>, <code>JMSTopic</code>, or <code>JMSDistributedQueue</code>, or <code>JMSDistributedDestination</code> with an existing name using <code>weblogic.Admin</code>, the Admin call returned an error, but still added a duplicated entry in the <code>config.xml</code> file.</p> <p>Now, the Admin call no longer adds a duplicated entry for <code>JMSQueue</code>, <code>JMSTopic</code>, <code>JMSDistributedQueue</code>, or <code>JMSDistributedTopic</code> in the <code>config.xml</code> file.</p>
CR219745	<p>Destinations can now be removed from a suspended migratable backend. As a result, when JMS servers are targeted to migratable targets, deleting and re-creating destinations no longer results in the following error:</p> <pre>weblogic.management.configuration.ConfigurationError: Error adding destination: instance already exists.</pre>
CR221262	<p>A JMS frontend connection and session instance leak no longer occurs.</p>
CR223209	<p>Expired messages with no expiration redirect policy defined are now dynamically removed from the store when they expire, so that store size no longer increases unnecessarily.</p>
CR224668	<p>A <code>java.lang.ClassCastException</code> is no longer thrown on server restart when JMS attempts to page-in and add a durable topic message to a consumer that was paged-out on server startup.</p>
CR225782	<p>JMS topic subscribers may temporarily stall while receiving messages if there is a delay on the request completion between client and server. This delay could occur due to network latency or an undetected peergone.</p> <p>To prevent a short term stall in message delivery, use the server side system property <code>-Dweblogic.jms.DisablePushEnvelope=true</code>.</p>
CR226919	<p>A JMS memory leak that was caused by a temporary destination <code>delete()</code> failure no longer occurs.</p>

## Resolved Problems for Service Pack 5

<b>Change Request Number</b>	<b>Description</b>
CR229963 CR237399	To avoid deadlock, the lock hierarchy for <code>browserCreate()</code> is modified to ensure that WebLogic Server is locking the destination before the session.
CR230297	A deadlock no longer occurs when distributed queues are added and deleted through shell scripts that use <code>weblogic.Admin</code> commands.
CR235404	Memory leaks were encountered if the producer was not explicitly closed before closing the client connection.  This problem has been fixed.
CR239533	Combo patch request of CR204981 and CR203841.

## JNDI

---

<b>Change Request Number</b>	<b>Description</b>
CR197207	<p>The Java client hung if the JNDI lookup was performed from a static block during the response from an EJB call.</p> <p>A new system property, <code>-Dweblogic.rjvm.t3.dispatchOnCompleteMessage</code>, needs to be enabled on the client side. When this property is enabled, the t3 request is processed on a thread other than the socket reader thread.</p>
CR210323	<p>When using a <code>javax.naming.CompoundName</code> to bind an object, WebLogic Server no longer throws a JNDI error if there are quotes and a dot in the object name.</p>

---

## JTA

---

<b>Change Request Number</b>	<b>Description</b>
CR048979	<p>A race condition no longer occurs when multiple enlistments of JTS connections attempt to initialize the resource at the same time.</p> <p>As a result, WebLogic Server no longer throws a <code>javax.management.InstanceAlreadyExistsException</code> while trying to get a JTS connection.</p>
CR186070 CR233670 CR231422	<p>The configured JTA transaction timeout is now applied to any EJBs that do database work at the time an application is deployed.</p>
CR205368	<p>A <code>java.util.ConcurrentModificationException</code> no longer occurs when WebLogic Server uses the JTA2PC debug flag and tries to print transaction information while another thread is trying to modify the hashtable used for local and global properties.</p>
CR211194 CR220787	<p>Calling the XA call sequence <code>XA.end (TMSUSPEND)</code>, followed by <code>XA.end (TMFAIL)</code> followed by an <code>XA.rollback</code> in the case of an MQSeries resource prevents memory leaks in MQSeries.</p>

---

<b>Change Request Number</b>	<b>Description</b>
CR212535 CR214984	WebLogic Server now ensures that the subcoordinator will roll back any resource it has locally if it is given an empty list from the coordinator.
CR215559	<p>When a resource involving a global transaction threw a Heuristic Hazard Exception, the transaction was declared completed regardless of the fact that all the subcoordinators and resources had not responded to the commit.</p> <p>Ideally, during a commit process, if one of the subcoordinating servers is down, the coordinator should not have committed till such time that it heard from all resources and subcoordinators. Because WebLogic Server did not allow all subcoordinators to complete the commit call, transactions remained pending even though the transaction had been declared committed.</p> <p>The problem has been resolved.</p>
CR215821	<p>When transactions span two domains and a commit is issued, the coordinating server is now able to reach the subcoordinating server and no longer throws the following exception:</p> <pre>Unable to get a direct or routed connection to...</pre>
CR235382	<p>A PROTO exception was thrown for an asynchronous rollback call because of the differing behavior of vendor resources. Vendors like MS SQL, DB2 and Tibco need to be unlisted before the rollback call, using the TMFAIL flag.</p> <p>This problem has been fixed.</p>

## JVM

Change Request Number	Description
CR210725	<p>When a client application spawned multiple threads to establish initialcontext and did the lookup from JNDI, the internal messages between the client and the WebLogic Server, sometimes used stale connection information on the server. Thus the server sent its response over a stale connection and the response never reached the client. This situation caused the client threads to hang indefinitely waiting for data and response from the server.</p> <p>WebLogic Server now synchronizes the dispatch of any messages with the connection management code. As a result, the client thread is prevented from being stuck indefinitely.</p>
CR218538	<p>A <code>JAVA LEVEL DEADLOCK</code> no longer occurs when the RJVM is being shut down.</p>
CR223414	<p>When multiple threads shared the same stub, sometimes WebLogic Server threw a <code>ConnectException</code> even when other servers were available in the cluster.</p> <p>Now, the server list in the cluster is refreshed at least once so the <code>ConnectException</code> is thrown only after none of the servers in the cluster are available.</p>
CR224102	<p>When using DNS name as the ListenAddress for the Administration Server, and restarting the Administration Server with a different IP address, the Managed Server could not re-connect to the Administration Server, and WebLogic Server threw a <code>PeerGoneException</code>.</p> <p>The host IP address stored in the Server is now reset if a <code>PeerGoneException</code> occurs.</p>

## Node Manager

Change Request Number	Description
CR128517	<p>Using the Administration Console, if you configure a machine, but do not configure Node Manager, and then you inquire about the state of the server, Node Manager is contacted by default.</p> <p>If you do not need the Node Manager, set the ListenPort to 0 in the machine configuration. This tells the server that the Node Manager is not configured, and it will not attempt contact.</p>

---

<b>Change Request Number</b>	<b>Description</b>
CR192334	<p>The <code>NodeManagerMBean</code> methods <code>getLogs()</code>, <code>getErrorLog()</code>, and <code>getOutputLog()</code> were not implemented and returned null.</p> <p>These methods were erroneously marked as public in the javadoc and are now marked private. They no longer appear in the public documentation.</p>
CR195322 CR208042	<p>After shutting down and restarting, <code>NodeManager</code> can now restart a Managed Server. In addition, <code>NodeManager</code> no longer fails to close sockets.</p>
CR206260 CR214012	<p>The Managed Server would not restart during the first <code>CheckIntervalSeconds</code> after <code>Node Manager</code> is restarted.</p> <p>To fix the problem, the <code>serverState</code> instance variable, which holds the Managed Server state, is now updated when the state of the server changes.</p>

---

# Operations, Administration, and Management

Change Request Number	Description
CR106432	To avoid incorrect configuration, if the config.xml file has multiple entries with the same name for the same MBean type, WebLogic Server no longer boots up and the parser displays an appropriate error message.
CR107552	WebLogic Server now throws a <code>MalformedObjectNameException</code> if ObjectNames specified on the command line are not correctly separated by the designated delimiter.  For example, the following ObjectName contains the illegal delimiter,  , and therefore, would result in a <code>MalformedObjectNameException</code> : <pre>SET -mbean mydomain:Name=sqlpool,Type=JDBCConnectionPool -property Targets mydomain:Name=OtherServer,Type=Server mydomain:Name=myserver ,Type=Server</pre>
CR108485	The new implementation now strictly enforces restrictions on attribute values. As a result, it is no longer possible to set a COMMO MBean attribute to a value that is lower than its minimum value and higher than the specified maximum.  If you set invalid values for attributes on any of the security providers, these values will now fail.
CR108714	Wlconfig no longer fails when queries return multiple object names and the return value is assigned to a property.  In addition, WebLogic Server now throws an <code>InvalidWebLogicObjectNameException</code> if the MBean parameter passed in resolves to multiple MBeans.
CR109300	The wl.Admin tool now accounts for the CommoMBeans and properly converts them to displayable Strings.
CR109386	weblogic.Admin VALIDATECLUSTERCONFIG no longer fails in cases where the cluster has no cluster address (default) or the server has no listen address (default).
CR109680	weblogic.Admin validateClusterConfig was output an error message even when the cluster was correctly configured.  Now, validateClusterConfig does not throw an error if the cluster name is an empty string for the server.

Change Request Number	Description
CR109743	<p>When a shutdown is issued on a server through JMX, WebLogic Server no longer throws the following exception:</p> <pre>weblogic.rjvm.PeerGoneException: ; nested exception is: java.io.EOFException</pre> <p>As a result, it is no longer necessary to capture and ignore <code>RemoteRuntimeExceptions</code> when shutting down the server.</p>
CR110861	<p>You can no longer use <code>weblogic</code> as the domain name.</p> <p>WebLogic Server now throws a clear configuration exception rather than a <code>NullPointerException</code> under the following circumstances:</p> <ul style="list-style-type: none"> <li>• If the domain's name is <code>weblogic</code> when WebLogic Server parses the configuration file</li> <li>• If you specified <code>-Dweblogic.Domain=weblogic</code></li> <li>• If you are creating a default configuration and attempting to use <code>weblogic</code> as the domain name.</li> </ul>
CR110865	<p>Now, WebLogic Server throws a <code>ConfigurationException</code> when two or more <code>ConfigurationMBeans</code> have the same name.</p>
CR120270	<p>The conversion utility now treats <code>JoltConnectionPool</code> as an <code>MBean</code> rather than as a startup class so that <code>JoltConnectionPool</code> is now converted properly.</p>
CR120684	<p>Now, the <code>weblogic.Admin validateclusterconfig</code> command can be executed without a username, password, or server URL.</p>
CR122793	<p>Because deprecated APIs are dependent on the Administration Server, it was necessary that the Administration Server be up to ensure a smooth shutdown of the Managed Server.</p> <p>Administration Server no longer needs to be up to perform a shutdown.</p>
CR124003	<p>It is now possible to set the HTTP Message Timeout to "-1" in the Administration Console.</p>
CR124169	<p>WebLogic Server no longer throws an Assertion error when shutting down or undeploying the WebLogic Messaging Bridge using JMX.</p>
CR125202	<p>The <code>chmod</code> command is no longer used on Windows. Now, <code>wlserver generateconfig</code> no longer fails on Windows if the <code>chmod</code> command is not used.</p>

<b>Change Request Number</b>	<b>Description</b>
CR129262	<p><code>ServerDebugMBean</code> interface debug attributes were not dynamic. Therefore, when the value was set on the Admin MBean, it was not propagated to the Config MBean on the Managed Server.</p> <p>Now, <code>ServerDebugMBean</code> attributes can be dynamically be set and un-set.</p>
CR132602	<p>The <code>weblogic.Admin</code> utility now allows access over the IIOP protocol when accessed through <code>-adminurl</code>.</p>
CR133869	<p>Now, <code>ServerMBean.getClusterRuntime</code> returns true for a server that is part of a cluster.</p>
CR137128	<p>Because increasing the array size and adding elements into an array are two different steps, the Array list was not synchronized. As a result, an <code>ArrayIndexOutOfBoundsException</code> was thrown when elements were added to Array list in two different threads.</p> <p>This exception is no longer thrown.</p>
CR175671	<p>You no longer get a <code>NullPointerException</code> when you use the <code>clz.getPackage()</code> method.</p>
CR175945	<p>Because the <code>running-managed-server.xml</code> file was being used to monitor the status of a Managed Server, the status was being displayed inconsistently and often incorrectly when viewed either from the Administration Console or from the command line.</p> <p>Now, when the Administration Server has discovery mode enabled, for each Managed Server entry in the <code>running-managed-server.xml</code> file, WebLogic Server attempts to determine if that server is running regardless of its status in the <code>running-managed-server.xml</code> file. As a result, Managed Servers are now being correctly discovered.</p>
CR177737	<p>The Administration Server failed to start if two JMS queues using the same JNDI name were in a cluster and if "Replicate JNDI Name In Cluster" was enabled.</p> <p>Now, JNDI name validation is done against tree destinations of all the concerned servers so that any conflict is detected before actual server startup.</p> <p>As a result, it is not possible to configure a destination that can restrain the server from starting, and an attempt at such configuration causes an error to be thrown.</p>

Change Request Number	Description
CR189644	<p>If logs had been rotated, the output of <code>View Server Log</code> in the Administration Console was not legible, and the order of the logs displayed was not correct because of the order in which file names were stored. There was also an inconsistency in the logs seen in Reverse chronological order and Forward chronological order. The output of <code>View Server Log</code> displayed the messages of the current log file as well as rotated log files, instead of just the current log file, for both reverse and forward orders.</p> <p>Now, the Console shows log messages only from the current log file and the contents shown in both forward and reverse chronological orders are the same.</p>
CR189763	<p>When a staged application was deployed to a server running in MSI mode, the server incorrectly created a staging directory.</p> <p>Now, when a staged application is deployed on a server running in MSI mode, WebLogic Server determines whether to use an existing staging directory or create a new one. As a result, there is no <code>InvalidAttributeValueException</code> when deploying to a server running in MSI mode where the server process does not have write access to the directory structure.</p>
CR191317	<p>If the Managed Server listen address was null and the Managed Server port was the same as the Administration Server port, WebLogic Server threw an error while starting the Managed Server if the Managed Server was started on a remote machine.</p> <p>Now, the Managed Server can be started on a machine other than the Administration Server machine in case its port is the same as the Administration Server port, even if its listen address is specified as null.</p>
<p>CR199214 CR193887 CR211395</p>	<p>Whenever the log was rotated, the next log rotation was calculated using the file time span and the rotation time set in the <code>LogMBean</code> which was always returning the time between the rotation time and 24:00. As a result, the logs were not getting rotated between 24:00 and the rotation time.</p> <p>Now, the next log rotation time is calculated by adding the file span to the previously set next log rotation time. After the server has been started, the next log rotation time is dependent only on the file time span.</p>
CR201967	<p><code>MBeanHomeImpl.getActiveDomain()</code> returned the <code>AdministrationDomainMBean</code>, which caused problems when the Administration server was down.</p> <p><code>MBeanHomeImpl.getActiveDomain()</code> now returns <code>Admin.getInstance().getActiveDomain()</code>, which is the local <code>ConfigurationDomainMBean</code> and does not rely on the Administration Server's availability.</p>

<b>Change Request Number</b>	<b>Description</b>
CR202036	On a Windows platform, the server log in the Administration Console now displays the last 2 characters of the line.
CR202059	WebLogic Server incorrectly persisted certain attributes of type <code>java.util.Properties</code> for COMMO MBeans. Upon retrieval, the values were incorrect. This issue has now been resolved by proper serialization and de-serialization.
CR203963	When trying to validate whether a user is locked out by using either the <code>ServerSecurityRuntimeMBean</code> or the <code>UserLockManagerMBean</code> on a Managed Server, the <code>isLockedOut</code> method and the <code>getUserLockoutTotalCount</code> method are now returning the correct results on the Managed Server.
CR205146 CR209670	When a Web Application is undeployed, the memory used in Permanent Generation is now freed upon a full garbage collection. As a result, the server no longer runs out of memory when the Web Application is repeatedly undeployed and redeployed.
CR208029	<p>WebLogic Server now throws an exception during bootup if both the old and new style attributes co-exist in the <code>config.xml</code> file.</p> <p>For example:</p> <pre>&lt;JoltConnectionPool ApplicationPasswordEncrypted= "{3DES}pPCVe/Lp17dc5WG18Ot3wg==" Name="MyJoltConnectionPool" Targets="cgServer" UserPassword="userpassword"/&gt;</pre> <p>is legal but</p> <pre>&lt;JoltConnectionPool ApplicationPassword= "appPassword"ApplicationPasswordEncrypte d= "{3DES}pPCVe/Lp17dc5WG18Ot3wg=="Name= "MyJoltConnectionPool " Targets="cgServer" UserPassword="userpassword"/&gt;</pre> <p>is illegal because it contains both <code>ApplicationPassword(old)</code> and <code>ApplicationPasswordEncrypted(new)</code> attributes specified that really represent the same entity.</p>
CR209218	The <code>CounterMonitor.stop()</code> method is no longer synchronized and there is no more deadlock.
CR209592	Security MBeans returned by the JMX query are no longer filtered out by the <code>MBeanServer</code> ; they are returned to the client.

Change Request Number	Description
CR209771	<p>The <code>TaskRuntimeMBean.printLog(pw)</code> function got a <code>NotSerializableException</code> when the method was invoked on a T3 client.</p> <p>The method was excluded from the public documentation, and the <code>TaskRuntimeMBean.printLog(pw)</code> method is no longer documented or available for public use.</p>
CR210425	<p>Now, the <code>&lt;wls&gt;</code> tag throws an exception when you try to stop the server without providing the username and password.</p>
CR211069 CR218084	<p>The Administration Server now makes adjustments to the encrypted attribute values when the Managed Server is on WebLogic Server 8.1 Service Pack 3 or earlier.</p> <p>As a result, when you upgrade an Administration Server to WebLogic Server 8.1 Service Pack 4 or later, you can start a Managed Server that is running on WebLogic Server Service Pack 3 or earlier.</p>
CR212225	<p>During a JMS stress test, WebLogic Server no longer throws the following exception: <code>NotSerializableException: weblogic.t3.srvr.T3Srvr</code></p>
CR213168 CR214825	<p>Certain private methods that accessed shared data have been synchronized to prevent inconsistent states during server startup.</p> <p>As a result, <code>ModelMBeanTypeMBean.operationCategory()</code> no longer throws a <code>NullPointerException</code> when starting multiple Managed Servers at the same time.</p>
CR214170 CR208509 CR221784	<p>The Console application javascript code was rewritten to work on all known versions of all browsers.</p>
CR214241	<p>When the <code>weblogic.Admin</code> tool <code>BATCHUPDATE</code> option was used, quotes were stripped from the file before processing.</p> <p>The logic on stripping quotes from batch files was changed to allow empty quotes to be passed. As a result, <code>wl.admin</code> now works the same in batch mode as in interactive mode.</p>
CR214288 CR236570	<p>Application deployment on a large cluster of nodes threw a <code>NullPointerException</code> on the Administration Server, of a synchronization problem in updating the configuration MBean list.</p> <p>This problem has now been fixed.</p>

Change Request Number	Description
CR214301 CR230260	<p data-bbox="357 418 1193 499">During the validation check of setting the <code>ErrorDestination</code> to a valid <code>JMSQueue</code>, the <code>JMSLegalHelper</code> code failed with a <code>NullPointerException</code> because the parent attribute had not been set.</p> <p data-bbox="357 517 1193 569">Now, if the parent attribute is not set, WebLogic Server tries to obtain the parent from the <code>WeblogicObjectName</code> for comparison and legal checks.</p> <p data-bbox="357 586 1193 638">As a result, WebLogic Server no longer throws a <code>NullPointerException</code> and still does the validation check.</p>
CR215114	<p data-bbox="357 670 1193 690">The JDBC pool commands usage strings have been updated to be consistent with code.</p> <p data-bbox="357 708 1193 821">RESET, ENABLE, DISABLE, EXISTS, TEST, REMOVE, SUSPEND, SHUTDOWN, RESUME, and DELETE use <code>-poolName &lt;poolName&gt;</code> to obtain the JDBC connection pool name for the specified operation. CREATE and DESTROY pool commands obtain the <code>poolName</code> through a positional parameter to the command line.</p>
CR216453	<p data-bbox="357 852 1193 904">The Monitor role could not see the deployment status on the Console because the required access permissions were not given.</p> <p data-bbox="357 921 1193 942">Now, monitor role can access the deployment status page.</p>
CR216513	<p data-bbox="357 973 1193 1025">When authorization errors occur, the Administration Console now shows a complete error message that explains the reason for the remote failure.</p>
CR219192	<p data-bbox="357 1057 1193 1109">When setting the <code>ErrorDestination</code> of the <code>JMSQueue</code>, a <code>NullPointerException</code> no longer occurs in the Administration Server.</p>
CR220015	<p data-bbox="357 1140 1193 1192">The <code>weblogic.Admin</code> utility returned an exit code of 0 when the START and RESUME commands failed.</p> <p data-bbox="357 1209 1193 1230">This problem has been fixed and now <code>weblogic.Admin</code> returns 1 as the exit code.</p>
CR221970	<p data-bbox="357 1262 1193 1373">WebLogic Server logger formats the log record before publishing it. When formatting bad records, any exception that occurred was not handled, which resulted in the log record not being published. After a certain number of such bad records, the logger deleted the handler, no records were published, and logging stopped.</p> <p data-bbox="357 1390 1193 1442">Now, WebLogic Server handles all exceptions during formatting and publishes the records. As a result, logging continues despite bad log records.</p>
CR223944	<p data-bbox="357 1473 1193 1525">The <code>weblogic.Admin BATCHUPDATE</code> command now returns accurate results when executed from the command prompt or as a batch file.</p>

## Resolved Problems for Service Pack 5

<b>Change Request Number</b>	<b>Description</b>
CR227566	<p>When generating custom providers, WebLogic Server generates <code>&lt;MBeanInterfaceName&gt;MBeanInfo.java</code> files that are used to generate the MBeanInfo files for the compiled MBeans.</p> <p>WebLogic Server now puts these source files in the same package as the MBean interface for which this class was generated. WebLogic Server no longer bundles these <code>*MBeanInfo.class</code> files in the provider's JAR file. As a result, the JAR files are much lighter.</p>
CR228265	<p>A timestamp formatted log file is now displayed in the Administration Console.</p>
CR232488	<p>The WebLogic Server command line tool displayed junk attribute values when it encountered encrypted data.</p> <p>This problem has been resolved.</p>

## Plug-Ins

<b>Change Request Number</b>	<b>Description</b>
CR128730	<p>When <code>HttpClusterServlet</code> sent a request to PRIMARY using a recycled connection, it did not determine that there was a failure to connect until it was too late to re-connect. As a result, <code>HttpClusterServlet</code> was never failing over to SECONDARY.</p> <p>Now, <code>HttpClusterServlet</code> checks to confirm whether the request was successfully sent to PRIMARY. Subsequently, if the request to PRIMARY was not sent successfully and <code>HttpClusterServlet</code> had used a recycled connection, <code>HttpClusterServlet</code> tries to create a new connection to PRIMARY. If the subsequent request also fails with an exception, <code>HttpClusterServlet</code> fails over to SECONDARY.</p>
CR175989	<p>The Apache server generated core dumps when using the worker (multi-threaded) option instead of the prefork (only multi-process) option.</p> <p>This was resolved by fixing the Locking and Unlocking logic.</p>
CR199080	<p>Now, when the plug-in fails to connect to WebLogic Server, <code>GETLASTERROR</code> is getting logged.</p> <p>Log messages have also been added to log any errors that occur while the plug-in is connecting to WebLogic Server.</p>
CR201397	<p>With <code>Idempotent=OFF</code>, the IIS proxy plug-in does not retry if it gets a <code>READ_TIMEOUT</code> exception after <code>WLIOTimeoutSecs</code> has elapsed.</p>
CR202898	<p>When a HEAD request is sent to an Apache plug-in, the headers in the <code>HttpResponse</code> are no longer missing the Content-Length header and the Content-Type header is no longer corrupted.</p>
CR205009	<p>Starting the Apache plug-in (revision 158) with root user no longer results in a core dump.</p>
CR205132	<p>Now, Apache 2.0.49 (with <code>mod_wl_20.so</code>) child processes are always terminated on HP-UX.</p>
CR205760	<p>If an <code>INSUFFICIENT_BUFFER</code> error was thrown while reading requests, then new memory was allocated. This memory was being accessed later in the method after it had been freed. Now, memory is freed at the end of the routine.</p>

<b>Change Request Number</b>	<b>Description</b>
CR206043	<p>When performing HTTP GET requests that were larger than 4096 bytes, the following error was seen in the <code>wlproxy</code> log files:</p> <pre>request too long: XXXX, max is 4096</pre> <p>This problem is resolved and the error is no longer thrown.</p>
CR206340	<p>The failover logic used during WebLogic Server shutdown has been fixed. WebLogic Server no longer returns HTTP 404 after SHUTDOWN sequence is initiated.</p>
CR207694 CR215550 CR217567	<p>Now, when "Expect: continue-100" is found in the request, NSAPI replies with "HTTP/1.1 100 Continue" in the header instead of in the response body.</p>
CR208303	<p>On AIX, the <code>errno</code> variable was not thread safe. To make the <code>errno</code> variable threadsafe, <code>-D_THREAD_SAFE</code> was added to the <code>Makefile.aix</code>.</p>
CR209383	<p>POST data is no longer getting chunked when <code>FileCaching</code> is set to OFF.</p>
CR209963	<p>Users could not access the welcome page of the application deployed on the backend WebLogic Server when <code>HttpClusterServlet</code> or <code>HttpProxyServlet</code> was used on the frontend WebLogic Server. The welcome page had to be appended to the URL while sending a request using the browser.</p> <p>Now <code>HttpClusterServlet</code> or <code>HttpProxyServlet</code> accesses the correct welcome page of the application.</p>
CR210497	<p>The <code>iisproxy128.dll</code> bits were not available in the installer because the path for the binary in the <code>build-native.xml</code> file for the installer was wrong.</p> <p>The path has been rectified and the <code>iisproxy128.dll</code> bits are now available in the installer.</p>
CR210899	<p>When the <code>DefaultFileName</code> property is set along with the <code>PathTrim</code> and <code>PathPrepend</code> properties, the final URI is no longer malformed.</p>
CR212235	<p>Multiple threads were updating the <code>ServerList</code> causing WebLogic Server to crash.</p> <p>To prevent this from happening, WebLogic Server now ensures that only one thread updates the global server list at a time.</p>
CR213200 CR237205	<p><code>GenericProxyServlet</code> now checks for EOF read from Input Stream before calling the <code>OutputStream.write()</code> method. As a result, an <code>ArrayIndexOutOfBoundsException</code> is no longer thrown.</p>

Change Request Number	Description
CR213809	<p>The older Apache plug-in used an undefined symbol <code>AP_NEED_SET_MUTEX_PERMS</code> on an HP-UX platform.</p> <p>This problem has now been fixed. Contact BEA Customer Support for the latest patch.</p>
CR216445 CR227315	<p>The size limitation for the response header has been removed. The NSAPI plug-in now allows response headers that are greater than 16k.</p>
CR216621	<p>Changes made by <code>mod_rewrite</code> to the URI are reflected only to <code>request_rec-&gt;uri</code>. Hence, the Apache plug-in now uses <code>request_rec-&gt;uri</code> by default.</p> <p>Also, a new property, <code>WLForwardUriUnparsed</code>, has been added. When it is set to ON, the Apache plug-in uses <code>request_rec-&gt;unparsed_uri</code> instead of <code>request_rec-&gt;uri</code>.</p> <p><b>Note:</b> The Apache plug-in with <code>WLForwardUriUnparsed</code> set to ON does not work correctly with <code>mod_rewrite</code>.</p>
CR216841	<p>Absolute URL was not converted to relative URL before processing it for <code>pathPrepend</code> and <code>pathTrim</code> when using an unparsed URI.</p> <p>Now, if the Apache plug-in detects an absolute URL, it first converts it to a relative URI before applying the <code>pathTrim</code> and/or the <code>pathPrepend</code> properties.</p>
CR218494 CR237656	<p>WebLogic Server no longer throws an error during graceful shutdown using a memory-based locking mechanism on Linux and HP_UX.</p>
CR220685	<p>WebLogic Server no longer retries using a new connection when the following are all true:</p> <ul style="list-style-type: none"> <li>• Idempotent is set to OFF.</li> <li>• A recycled connection is being used.</li> <li>• The backend server is not responsive.</li> </ul>
CR222702	<p>Apache Web server no longer generates huge log files under load if <code>APLOG_NOTICE</code> log level is used within the plug-in.</p>
CR222855	<p>Now, the latest proxy plug-in for SunOne 6.1 serves the <code>ErrorPage</code> before <code>WLIOTimeoutSecs</code>.</p>
CR228256	<p>In a clustered environment, WebLogic Server no longer throws a <code>java.net.SocketTimeoutException</code> when a client with WebDAV implementation accesses the Managed Server through the <code>HTTPClusterServlet</code>.</p>

---

<b>Change Request Number</b>	<b>Description</b>
CR229338	Now, the plug-in infrastructure logs errors when it fails to open the default log file and when it fails to resolve the hostname.
CR229990	To avoid a delay of 200 milliseconds when receiving the response from the ISAPI plug-in, the Nagle algorithm has been turned off.
CR232588	The <code>CookieName</code> parameter in the WebLogic Apache module overrides the <code>CookieName</code> parameter in Apache <code>mod_usertrack</code> module.  Replace the <code>CookieName</code> parameter with <code>WLCookieName</code> for a WebLogic Apache module.
CR235434	When the NSAPI plug-in with Sun One 6.1 is used, the HTTP response header no longer records two instances of the "Date" field.
CR237406	On the HP-UX platform, the installer could not install the 128-bit SSL version of the Apache plug-in.  The plug-in is now included as part of the installer.

---

# RMI

<b>Change Request Number</b>	<b>Description</b>
CR185615	<p>In order to allow WebLogic Server to inter-operate with WebSphere, the default values for Common Secure Interoperability Version 2 (CSIv2) deployment descriptors are no longer hardcoded. The default values are now configurable using MBeans. Use <code>IIOPMBean.setSystemSecurity</code> to change the defaults.</p>
CR197396	<p>Performance is improved because the CPU load on machines no longer increases to 100% while server instances call <code>addIndirection</code> for IIOP Outbound responses to clients calling an EJB with RMI/IIOP.</p>
CR198316	<p>The <code>equals</code> method of <code>ClientRuntimeDescriptor</code> did not consider the case when a custom classloader was the <code>contextClassLoader</code> and the interfaces were loaded by the custom classloader. As a result, too many instances of <code>ClientRuntimeDescriptor</code> were created, causing the CPU usage to peak.</p> <p>The <code>equals</code> method of <code>ClientRuntimeDescriptor</code> now checks for the custom classloader, which results in fewer instances of <code>ClientRuntimeDescriptors</code> and less CPU usage.</p>
CR201444	<p>WebLogic Server no longer throws a <code>NullPointerException</code> when accessing an EJB over IIOP.</p>
CR206947	<p>Marshalling large objects over IIOP was slower than over T3.</p> <p>A system property, <code>-Dweblogic.iiop.useJavaSerialization</code>, was added to WebLogic Server. When set to <code>True</code> on the server side, this property uses java serialization and thereby improves the performance when marshalling large objects. The property can also be set on the <code>IIOPMBean</code>. When it is set on the <code>IIOPMBean</code> the property is called <code>UseJavaSerialization</code>.</p> <p>For large objects, when this property is set, performance over IIOP is equivalent to performance over T3. However, if remote objects are embedded inside the serializable objects returned by the remote calls, WebLogic Server ignores the property.</p>
CR208687	<p>There was an interoperability issue with WebSphere when using EJBs in a Web Application for lookup. A <code>java.lang.ClassCastException</code> was thrown as there was an incorrect stub generation and serialization of nodes in a cluster environment.</p> <p>This problem has been resolved.</p>
CR213332	<p>WebLogic Server thin client no longer receives <code>NULL</code> as the return value of an EJB call that returns a dynamic proxy object.</p>

Change Request Number	Description
CR213643	<p>When using the patch for CR202488 with the administration port enabled, WebLogic Server no longer throws the following exception:</p> <pre>PeerGoneException: Duplicate Connection and EOFException</pre>
CR215851	<p>When using IIOP protocol, a <code>TransactionRolledBackException</code> from WebLogic Server resulted in a <code>ClassCastException</code> on the client side.</p> <p>Now, the method is properly invoked depending on the type of transaction. As a result, the <code>ClassCastException</code> is no longer thrown.</p>
CR217023	<p>In clustered environments, WebLogic failover logic left a user on the thread stack, which can cause certain side effects apart from a minor memory leak. WebLogic Server now ensures that this user is popped from the stack and the context is properly closed so that all associated objects get garbage collected.</p>
CR217025	<p>If you were using <code>JavaSocketMuxer</code> on the server and had an incorrect classpath configuration in which the client knew about the classes but the server did not have those classes in its classpath, an EJB invoked from the client sometimes failed with no exception on the server making it difficult to debug the root cause of the issue.</p> <p>Now, WebLogic Server properly reports the exception.</p>
CR218848 CR204958	<p>WebLogic Server can now timeout a client-side synchronous RMI call to EJBs by exposing the <code>remote-client-timeout</code> parameter through the EJB descriptors. The client gets an exception if the call does not complete within the specific timeout duration.</p> <p>The timeout is implemented for the IIOP protocol. It was fixed for T3 through CR176676 in WebLogic Server 8.1 Service Pack 4.</p>

# Security

<b>Change Request Number</b>	<b>Description</b>
CR124239	It is now possible to search on an attribute immediately after it has been added to an entity in an embedded LDAP server using an external tool or API.
CR125681	Principal validator now validates the principals properly and no longer throws exceptions during validation.
CR177523 CR224395	When proxying a connection, WebLogic Server was validating the proxy hostname rather than the target system hostname, which resulted in a validation error. Now, WebLogic Server validates the target system hostname.
CR181752	Refer to the security advisory information available at: <a href="http://dev2dev.bea.com/pub/advisory/146">http://dev2dev.bea.com/pub/advisory/146</a>
CR186439 CR214220	When a domain with a cluster is started through the Administration Console, the locking mechanism for the LDAP server is now working properly. As a result, Managed Servers are no longer hanging during startup.
CR194529	Refer to the security advisory information available at: <a href="http://dev2dev.bea.com/pub/advisory/152">http://dev2dev.bea.com/pub/advisory/152</a>
CR201978	When WebLogic Server discovered that the embedded LDAP index file was corrupted, the error was not always passed up to the calling code. As a result, the calling code continued to process requests, based on the faulty index file, which triggered more exceptions. The log file no longer contains records for the searches on faulty indices. The incorrect index file is now correctly handled.
CR206421	Refer to the security advisory information available at: <a href="http://dev2dev.bea.com/pub/advisory/154">http://dev2dev.bea.com/pub/advisory/154</a>
CR206881	WebLogic Server does not allow the value for the <code>SSLLoginTimeoutMillis</code> to be set to 0 in the <code>config.xml</code> file. WebLogic Server documentation no longer states that a value of 0 disables the attribute.
CR210229	Calling the <code>ServletAuthentication.weak()</code> method no longer results in an <code>EmptyStack</code> exception.

Change Request Number	Description
CR210310 CR231457	The CA certificates that had the path length constraint field omitted in their basic constraints extension were not recognized as CA certificates. As a result, certificate chains containing such certificates failed certificate validation during SSL connection. Now, such certificates are interpreted correctly as CA certificates with no path length constraint.
CR210734	<p>Support for the following two AES cipher suites has been added:</p> <ul style="list-style-type: none"> <li>• TLS_RSA_WITH_AES_128_CBC_SHA</li> <li>• TLS_RSA_WITH_AES_256_CBC_SHA</li> </ul> <p>In order to use TLS_RSA_WITH_AES_256_CBC_SHA cipher suite, you need to install JCE unlimited strength jurisdiction policy files instead of the policy files that are shipped with the JDK by default.</p>
CR211656	<p>Permission statement parsing was not handling wildcard characters and escaped quotes except in one specific use case.</p> <p>To fix this, the parsing routine was changed so that it can handle any valid combination of wildcarding and quote escaping. Any possible valid combinations of clauses in the <code>ra.xml</code> are now being handled without any error.</p>
CR211853	There is no longer a socket leak when WebLogic Server makes a SOAP call over HTTPS to Sun One Identity Server.
CR212348 CR231517	<p>The <code>listGroupMembers()</code> method's SSPI call was aborting with an exception if one of the group member's Distinguished Names was invalid.</p> <p>Now, the <code>listgroupmembers()</code> method shows the valid groups and ignores the members whose Distinguished Names cannot be validated.</p>
CR213399 CR175135	<p>A custom object extending <code>UserInfo</code> could not be passed into <code>InitialContext</code> for authentication, which broke backwards compatibility.</p> <p>The "Custom Object Authentication Enabled" attribute was added to the Security Mbean and the Compatibility Advanced tab on the Administration Console so that if a user passes a custom object that extends <code>UserInfo</code> for authentication, it will be taken care of. This attribute is disabled by default because it affects performance.</p>
CR214372	The token from the request header is being handled correctly when it is used to assert identity. In addition, if certificates are supplied, but there is no identity asserter that supports them, then WebLogic Server throws an exception.
CR214423	There is no longer a memory leak with <code>weblogic.net.http.HTTPSClient</code> .

<b>Change Request Number</b>	<b>Description</b>
CR215121	The <code>socket.close()</code> method was fixed to prevent the file descriptors from increasing in number and thus causing a memory leak.
CR215515	Some versions of Internet Explorer hang when WebLogic Server sends 0 length packets. Now, you can disable sending 0 length packets by adding the public property, <code>-Dweblogic.security.SSL.DisableSSLEmptyRecords=true</code> . Internet Explorer does not hang if this property is enabled. The default value of this property is <code>False</code> .
CR215718	The multiple step identity assertion interfaces are now available in the WebLogic Server security framework.
CR215837	When a client tries to start an application by Java WebStart using T3s to connect to the server, the client no longer gets a missing license exception when it tries to connect.
CR215960 CR237210	JSAFE native library support has been added. Since WebLogic Server 8.1 SP5 ships only the FIPS jsafe jar, please contact the support service of the vendors, for other native libraries.
CR218149	Custom exceptions thrown by Custom Realms are now propagated up the call stack by WebLogic Server.
CR218240	Refer to the security advisory information available at:
CR230119	<a href="http://dev2dev.bea.com/pub/advisory/122">http://dev2dev.bea.com/pub/advisory/122</a>
CR230439	
CR233424	
CR219417	<code>ClearOrEncryptedService</code> no longer causes a <code>NullPointerException</code> when executed from a JMX remote client.  <code>ClearOrEncryptedService</code> is only available on the server-side. WebLogic Server now throws an exception if it is instantiated on the client.
CR220061	JMS topic receivers stopped receiving messages when the network cable was unplugged from another machine where other JMS receivers were installed.  To fix this, changes were made to the outputstream wrapper class so that the new socket times out properly.
CR222113	Using the Administration Console, you can now define security policies for the methods of your Web Service to allow access to specific roles.
CR224395	The Certicom code no longer fails with a <code>NullPointerException</code> .

Change Request Number	Description
CR226331	WebLogic Server 8.1 SP4 now returns the complete client certificate information to the <code>UserNameMapper</code> implementation.
CR227505	While doing the dynamic group search, WebLogic Server no longer abandons the search when the list contains an invalid URL. Instead, it continues with the valid URLs.
CR227734	New policies have been added to enforce the <code>JDBCConnection</code> operation permissions at the JDBC layer level. As a result, if you grant access to a resource by defining policy that the security providers enforce, access is no longer denied if it gets blocked at the JMX policy layer.
CR229605	Now, <code>UseRetrievedUserNameAsPrincipal</code> and <code>UserNameAttribute</code> are both working correctly when <code>SAMAccountName</code> defines the user attribute.
CR229976	A failed LDAP Connection from the pool is now silently retried with a new connection.
CR232551	Entitlement language characters such as hyphen and space were not supported in the Administration Console, WebLogic Server Scripting Tool, or while calling MBean APIs. This problem has been resolved by encoding the entitlement language syntax with the current entitlement expression syntax.
CR234477	An <code>InvalidParameterException</code> was thrown while deploying a resource adaptor when the associated deployment descriptor contained permission statements that used wildcard characters or escaped quotes. This problem has been resolved.
CR234765	Size of the EmbeddedLDAP files continued to increase even after deleting data. Now all files are re-written and data compressed when it reaches the configured size limit. The size is set by the <code>weblogic.security.ldap.maxSize=&lt;max bytes&gt;</code> system property.
CR234996	There was a timing issue when connecting a restarted Managed Server to an Administrative Server, over a secure port which was not yet initialized. This problem has been fixed. Any connection failure or <code>NotYetInitializedExceptions</code> thrown because of a secure connection can be ignored.
CR240524	Combo patch request of CR239321 and CR212348.

## Servlets and JSPs

Change Request Number	Description
CR085091	A custom error page, if defined, is now served if the request is a conditional GET (Is-Modified-Since header set) for a protected resource.
CR136682	You can now enable form-based authentication over https even if cookies are disabled in the browser. However, enabling this feature forces the browser to display the Authorization Cookie as part of the URL. To enable this feature, add the following flag to the Weblogic start script: <code>-Dweblogic.http.AuthCookieURLRewritingEnabled=true</code>
CR174837	The unnecessary cast to <code>weblogic.servlet.jsp.PageContextImpl</code> in the generated JSP has been removed. As a result, custom <code>PageContext</code> implementations are now usable.
CR183350	<p>If the <code>local-path</code> used with the virtual-directory-mapping was a relative path (not an absolute file path), it was resolved differently in WebLogic Server 7.0 and WebLogic Server 8.1.</p> <p>Now, in WebLogic Server 8.1 and later, the relative path is first resolved against the Web Application docroot directory and if that directory does not exist, it is resolved against the server's root directory.</p> <p><b>Note:</b> The server's root directory can be overridden by the <code>-Dweblogic.RootDirectory</code> property.</p> <p>When migrating the Web Application from WebLogic Server 7.0 to WebLogic Server 8.1 with the following entry, the entry needs to be modified:</p> <pre>&lt;virtual-directory-mapping&gt;   &lt;local-path&gt;./&lt;/local-path&gt;   &lt;url-pattern&gt;/portal/*&lt;/url-pattern&gt; &lt;/virtual-directory-mapping&gt;</pre> <p>In this case, <code>local-path</code> is referring to the current directory. First, the path <code>./</code> will be resolved against the Web Application document root. Because this always exists, it will not be resolved against the server root directory and the files from virtual directory will not be found at the runtime. To work around this problem, create a new directory with a name <code>&lt;newDir&gt;</code> under the server's root directory, move the <code>virtualDirectory</code> files into this new directory, and change the <code>local-path</code> to <code>./&lt;newDir&gt;</code>.</p>

Change Request Number	Description
CR189815	<p>If a change was made in the <code>HttpServletContext</code> that extended <code>ActionForm</code> in Struts framework, WebLogic Server threw a <code>ClassCastException</code> while accessing it.</p> <p>Now, when the servlet is reloaded, any non-serializable attribute (stored in <code>HttpServletContext</code>) that is loaded using the servlet or its child classes is removed from <code>HttpServletContext</code>. As a result, the WebLogic Server no longer throws a <code>ClassCastException</code> while accessing <code>HttpServletContext</code>.</p>
CR190817	<p>The <code>save-sessions-enabled</code> attribute now works when <code>context-root</code> is set to: <code>&amp;quot;/&amp;quot;</code>;</p>
CR202054	<p>When Apache Cocoon is used to present icons to the browser, performance is faster and better.</p>
CR202217	<p><code>WLObjectOutputStream</code> now writes the annotation and <code>WLObjectInputStream</code> reads the annotation so that the proper classloader is set to load the proxy objects. As a result, WebLogic Server no longer throws a <code>ClassNotFoundException</code> when storing proxy objects in a session.</p>
CR202495	<p>Refer to the security advisory information available at:</p>
CR236810	<p><a href="http://dev2dev.bea.com/pub/advisory/130">http://dev2dev.bea.com/pub/advisory/130</a></p>
CR239392	
CR206002	
CR205986	<p>Calling <code>HttpSessionBindingListener.valueUnbound()</code> while WebLogic Server code synchronized on the session (due to a call to <code>session.invalidate()</code>) caused a deadlock.</p> <p>Now, if <code>session.invalidate()</code> is called from two different threads, one of them receives an <code>IllegalArgumentException</code>, and if the same thread calls <code>session.invalidate()</code> again, the call is simply ignored.</p>
CR207215	<p>The Apache Web server too much memory if the response to an HTTP/1.0 request contained the "Accept-Ranges" header.</p>
CR222689	<p>WebLogic Server now only sends the "Accept-Ranges" header to the response to an HTTP/1.1 request.</p>
CR207481	<p>When the incoming JSP request URI is transformed in the filter, the Web container no longer generates duplicate JSPStubs for the same transformed URI and therefore no longer leaks memory.</p>
CR212670	
CR214349	

<b>Change Request Number</b>	<b>Description</b>
CR207643	When set to True, a new property, <code>-Dweblogic.allowCommasInNetscapeCookie</code> , allows commas as valid separators in a Netscape cookie. The default is False.
CR208931 CR213764	The internal flag is now set correctly if the same socket is re-used to serve a chunked-encoded request followed by a content-length request. As a result, the content-length request is no longer treated as a chunked request.
CR209626	Use of the inner static class with the <code>&lt;jsp:usebean&gt;</code> tag was broken because the file name should have had a period (.) to refer to the file, but instead used a dollar sign (\$).  To fix the problem, a period (.) was added for an inner class, so the new code is now defined as in the following example:  <code>new SubClassTest.Data()</code>
CR210386	Port 80 can now be used for the HTTPS protocol.
CR211039	Session invalidation no longer causes an <code>EJBException</code> for local interfaces stored in the HTTP session.  Session cleanups are now carried out using the classloader of the Web Applications in which they are loaded.
CR211133	Now, WebLogic Server honors the <code>FileCount</code> property when <code>FileName</code> contains <code>SimpleDateFormat</code> .
CR211497	The JSP compiler did not trim the extra white spaces from the generated page. The JSP compiler grammar was modified to adhere to the JSP 1.2 specification when processing white space in the JSP page. As a result, the size of the generated page is reduced and the overall page rendering is now improved.
CR211927 CR244881	A classloader mismatch error occurred when the replicated HTTP session had EJB stubs that were not packaged with the Web Application.  This problem has been resolved.
CR212058	Query string is no longer modified when the <code>getRequestDispatcher</code> method is called on <code>ServletRequestImpl</code> .
CR212871	WebLogic Server no longer throws a stack overflow exception regardless of how many JSPs are in the Web Application.

Change Request Number	Description
CR213426	<p>When a client application using a <code>HttpClusterServlet</code>, executed a POST using chunked transfer encoding, a Managed Server in a cluster returned the following:</p> <p><code>"501 Not Implemented"</code></p> <p>WebLogic Server now handles the chunked request correctly.</p>
CR214104	<p>SessionID generated with reserved keyword in the first 4 letters was impacting performance.</p>
CR222683	<p>The sessionID generation logic was modified to fix this issue.</p>
CR228624	
CR214245	<p>The default Web Application from the <code>defaultHttpServer</code> was getting undeployed when any other unrelated Web Application was undeployed from the virtual host.</p> <p>Code was fixed to resolve this issue. If any Web Application is undeployed from a virtual host, the default Web Application from the <code>defaultHttpServer</code> will not be undeployed.</p>
CR214988	<p>Logic was added to remove from memory all <code>SessionData</code> objects whose corresponding Session IDs are non-existent in the database. This cleanup of orphaned <code>SessionData</code> happens once in every 10th run of the <code>invalidateSession</code> trigger.</p> <p>As a result, there is no longer a memory leak when using JDBC Session Persistence in a clustered environment.</p>
CR217051	<p>Setting <code>fileCaching=false</code> in the <code>web.xml</code> file is now honored for chunked post requests.</p>
CR217213	<p>When reading large files, the <code>sun.io.MalformedInputException</code> was thrown if the command line options <code>'-Dfile.encoding=UTF-8'</code> and <code>'servletRequest.setCharacterEncoding("UTF-8")'</code> were used together.</p> <p>This problem has been resolved.</p>
CR219602	<p>WebLogic Server no longer throws an <code>IOException</code> (<code>java.net.SocketException</code>) when the socket connection gets reset for whatever reason, such as the browser closing when the page is being processed or the page refreshing when the page is being processed.</p>
CR236224	
CR220782	<p>When WebLogic Server throws a compile error of the servlet-mapped JSP file when a JSP registered as a servlet uses the <code>&lt;servlet&gt;</code> element in <code>web.xml</code> file, the <code>JSPStub</code> corresponding to this JSP is no longer removed.</p> <p>As a result, when the compilation error of the servlet-mapped JSP file is corrected, this JSP is compiled again and served. Because the mapping exists in the URI map for the required <code>JSPStub</code>, WebLogic Server no longer throws a 404 Page not found error.</p>

<b>Change Request Number</b>	<b>Description</b>
CR220806	The <code>weblogic.http.session.maxConcurrentRequest</code> property has been added to limit the number of concurrent requests for a session. If the number of concurrent requests for a given session exceeds the specified value, the servlet container will start rejecting requests. By default, this property is set to -1, which indicates the servlet container does not impose any restrictions.
CR225116	In case of nested or concurrent invalidation of sessions, WebLogic Server occasionally threw <code>EmptyStackExceptions</code> , due to the internal pops on the user stack.  This problem has been resolved.
CR227702 CR239269	Now, WebLogic Server serves the Welcome file from a directory defined in the <code>weblogic.xml</code> file as a virtual directory and residing anywhere on the system (not necessarily under the application root).
CR228381	WebLogic Server threw an <code>ApplicationException</code> when deploying the <code>ConsoleExtensibilitySample</code> Web application on WebLogic Server 8.1 SP4 version.  This problem has been resolved.
CR228812	WebLogic Server no longer throws an <code>Assertion Error</code> when an HTTP POST request to an unsecured resource is forwarded ( <code>RequestDispatcher.forward</code> ) to a secured resource.
CR229006	When a page directive is defined in an including resource with <code>session="false"</code> , <code>weblogic.jspc</code> no longer ignores the <code>session="false"</code> setting.
CR229062	Now, when Servlet Filter Initialization fails, undeploying the application no longer fails.
CR229334	Refer to the security advisory information available at: <a href="http://dev2dev.bea.com/pub/advisory/139">http://dev2dev.bea.com/pub/advisory/139</a>
CR229577	In <code>FormBasedAuthentication</code> , the requested URL was not getting properly stored, both before and after the user was being authenticated using the form-based login page.  To store the complete URL (including the http or https protocol scheme) of the original request, the following parameter was added to the <code>weblogic.xml</code> file:  <pre>&lt;container-descriptor&gt; &lt;retain-original-url&gt; true &lt;/retain-original-url&gt; &lt;/container-descriptor&gt;</pre>

Change Request Number	Description
CR230285	<p>Now, when WebLogic Server is overloaded, a <code>weblogic.util.UnsyncCircularQueue\$FullQueueException</code> is no longer thrown. Instead, the Web container handles this condition and sends a Service not available response to the client.</p>
CR232639	<p>Some HTTP sessions were not invalidated after the timeout period expired, causing an out of memory error.</p> <p>All the sessions now expire on timeout.</p>
CR232735	<p>In some cases, the http response time was slow and delayed as each response was delivered in two separate TCP packets, regardless of its size.</p> <p>Except for certain exceptional scenarios, the http response is now sent as a single TCP packet by default. However, a large http response can also be sent in two packets, if the message size is larger than the value of the <code>CHUNKSIZE</code> variable. You can configure the <code>CHUNKSIZE</code> variable to determine the size of a large response message.</p>
CR234441	<p>WebLogic Server allowed the user to pass name as null in <code>ServletRequest.setAttribute</code>.</p> <p>WebLogic Server now throws an <code>IllegalArgumentException</code> when a null entity is passed as name for <code>ServletRequest.setAttribute</code>.</p>
CR235025	<p>System property <code>DynamicServerList=OFF</code> now works in <code>HttpClusterServlet</code>.</p>
CR235555	<p>Contents of a large JSP file are no longer garbled because the <code>jspc</code> compiler no longer ignores <code>pageEncoding</code>.</p>
CR240564	<p>Combo patch request of CR236810 and CR229949.</p>
CR240572	<p>Combo patch request of CR236810 and CR230070.</p>

# SNMP

<b>Change Request Number</b>	<b>Description</b>
CR204059	WebLogic Server threw a <code>java.lang.IllegalArgumentException</code> when retrieving the SNMP information for <code>OID .1.3.6.1.4.1.140.625.650</code> . Under the covers, the value retrieved from the JMS runtime could not be converted back to the interface type in MIB. This problem has been resolved.
CR206283	LogBroadCaster handlers did not receive trap notifications when the domain log flag was set to false.  This problem has been resolved.
CR206632	When the BEA SNMP Agent came up, it initialized all the server states in the domain to False by default. During the next refresh cycle, when it actually checked the status on each server, the status for servers that had already been running when the Administration Server was booted was True. The SNMP agent then issued a trap because of a state change. Because the servers had been running the whole time, the SNMP agent should not have issued the trap.  Now, WebLogic Server initializes the Managed Server states correctly when the Administration Server boots up so that these false traps are not sent out.
CR209397	New EJBRuntime attributes were added to the MBean and the Management Information Base (MIB), but the EJBRuntime Handler was not updated to retrieve these attribute and return them.  Code has been added to the EJBRuntime Handler to return the values for these runtime attributes. As a result, SNMP now works correctly when <code>EJBPoolRuntimeAccessTotalCount</code> is used.
CR216306	The <code>jmsPooledConnectionRuntime</code> handler can now find the attributes in the Management Information Base (MIB) and there are no more problems running <code>snmpwalk</code> .
CR222114	The enterprise OID for the SNMP <code>coldStart</code> trap was incorrect. It has been changed to <code>".1.3.6.1.4.1.140.625"</code> .

## Tools

<b>Change Request Number</b>	<b>Description</b>
CR098504	The value <code>replicated_if_clustered</code> has been added to the drop down menu in Marathon, so that you can now set <code>replicated_if_clustered</code> for a session-param.
CR183190	The DB2 driver was improved to prevent the <code>java.sql.SQLException</code> .
CR203256	EJBGen now generates the proper DD entries for transaction attributes on local home and create methods.
CR210252	EJBGen now recognizes <code>max-beans-in-free-pool</code> and <code>initial-beans-in-free-pool</code> for entity beans.
CR212713	Calling <code>setFetchSize()</code> while using the Oracle thin driver was corrupting data. Oracle delivered a patch thin driver to fix this problem. Call BEA Customer Support to obtain the patch.
CR216571	WebLogic Builder now supports adding <code>resource-env-ref</code> and <code>res-env-ref</code> elements.
CR220246	WebLogic Server no longer prints out default values for <code>&lt;remote-client-timeout&gt;</code> .
CR227292	Now, when using the WebLogic Server java Utility DDInit tool, WebLogic Server correctly writes the context-root tags in the <code>application.xml</code> file without the <code>.war</code> extensions.
CR230460	Marathon now fully supports editing <code>resource-env-description</code> elements.
CR235970	WebLogic Builder took a long time to save very large EAR files. As a result of the extended save time, the application appeared to hang. Inefficiencies in the saving mechanism of EAR files have been fixed, so large files are now saved much faster.

## Utilities

<b>Change Request Number</b>	<b>Description</b>
CR199899	<code>weblogic.ejb</code> no longer fails when there is a \$ symbol in the interface class name.
CR200530	Using the <code>java weblogic.marathon.ddinit.EarInit</code> method in a multi-user environment resulted in access issues for the temporary directory created by the method. Only one user could write to that directory at any given time.  You can now specify your own temporary directory using the <code>-Dmarathon.tmpdir</code> flag when you invoke the <code>weblogic.marathon.ddinit.EarInit</code> method.
CR229018	Now, <code>DDconverter</code> supports empty query strings and successfully automates the migration of deployment descriptors from EJB 1.1 to EJB 2.0.

## Web Services

<b>Change Request Number</b>	<b>Description</b>
CR181775 CR229016	WebLogic Server generated the type dynamically when <code>xmlType</code> was not an element of the known schema during deployment. As a result, the element names from the original imported schema were not preserved in the generated WSDL file.  A system property, <code>-Dweblogic.webservice.noDynamicTypeGeneration</code> , has been added to WebLogic Server which, when set to true, preserves the original element names.
CR185173	When calling into a WebLogic Server 7.0.2 Web Service from a WebLogic Server 6.1.4 dynamic client, running the <code>webserviceclient+ssl.jar</code> through the <code>VersionMaker</code> utility, and then using the modified jar file in the WebLogic Server 6.1.x environment, WebLogic Server no longer throws a <code>java.lang.NoClassDefFoundError</code> .
CR189437	It is now possible to send a simple text file attachment from the test client in BEA WebLogic Workshop® to a Web Service created using WebLogic Workshop.
CR189437	WebLogic webservice client is now able to send a simple text file attachment.

Change Request Number	Description
CR189884	<p>There was a problem handling the inclusive namespaces prefix list in the C14N implementation, resulting in failure of signature verification.</p> <p>This problem has been resolved.</p>
CR198708 CR212958	<p>A Web Service client running from an applet no longer fails with an <code>AccessControlException</code>.</p>
CR202714 CR235349	<p>When importing an XML Schema as the default namespace, WebLogic Server no longer adds unnecessary namespaces in the SOAP request.</p>
CR204082	<p>A WebLogic Web Services client no longer throws a <code>StackOverflow</code> exception while deserializing <code>XSDAnyType</code>.</p>
CR208435	<p>Signature verification no longer fails for Web Services security draft 12 specification on WebLogic Server.</p>
CR208557	<p>When verifying a WSEE 1.0 signed soap message, WebLogic Server no longer throws a <code>java.lang.Exception</code>.</p>
CR208927	<p>If the response object is a custom type and contains an instance variable <code>XXX</code> that is of type <code>boolean</code>, the servicegen ant task generates classes that have the method <code>isXXX()</code> in addition to <code>getXXX()</code>.</p>
CR210221	<p><code>BusinessEntity CategoryBag</code> is no longer being overridden by <code>BusinessService CategoryBag</code>.</p>
CR210455	<p>The WebLogic Web Service client no longer throws an <code>ArrayIndexOutOfBoundsException</code> when a user-defined exception that extends <code>org.apache.commons.lang.exception.NestableException</code> is thrown by the webservice method.</p>
CR210918	<p>WebLogic Server threw the following exception on the client side:  <code>javax.xml.rpc.soap.SOAPFaultException: One or more references failed to validate, because of the addition of a spurious default namespace to an unqualified xml attribute.</code></p> <p>To resolve the problem, the code generation tool was fixed to generate the <code>XMLName</code> with just the <code>localName</code> (no <code>namespaceUri</code>) when the attribute is unqualified.</p>

Change Request Number	Description
CR211670	<p>The Ant tasks supporting Web Services development fail unless the <code>webservices.jar</code> is included in the classpath.</p> <p>The problem was resolved by adding <code>webservices.jar</code> to the classpath in <code>WL_HOME/common/bin/commEnv.sh</code> script file.</p>
CR211994	<p>The WebLogic Web Service client now sends back all cookies in multiple set-cookie headers from the server.</p>
CR213425	<p>WebLogic Server no longer throws a <code>weblogic.xml.schema.model.XSDEException</code> when <code>clientgen</code> is used against <code>wsdl</code> with imported namespace in the schema.</p>
CR216574 CR235981	<p>The <code>autotype</code> Ant task does not comply with the JAX-RPC specification if the XML Schema data type (for which it is generating the Java representation) has all the following characteristics:</p> <ul style="list-style-type: none"> <li>• The data type is a <code>complexType</code>.</li> <li>• The <code>complexType</code> contains a single sequence.</li> <li>• The sequence contains a single element with <code>maxOccurs</code> greater than 1 or unbounded.</li> </ul> <p>The following example shows such an XML Schema data type:</p> <pre>&lt;xsd:complexType name="Response"&gt;   &lt;xsd:sequence &gt;     &lt;xsd:element name="code" type="xsd:string" maxOccurs="10" /&gt;   &lt;/xsd:sequence&gt; &lt;/xsd:complexType&gt;</pre> <p>The <code>autotype</code> Ant task maps this type of XML Schema data type directly to a Java array of the specified element. In the previous example, the <code>autotype</code> Ant task maps the <code>Response</code> XML Schema data type to a <code>java.lang.String[]</code> Java type. This is similar to the type of mapping that .NET does.</p> <p>The JAX-RPC specification, in turn, states that this type of XML Schema data type should map to a Java array with a pair of setter and getter methods <i>in a JavaBean class</i>. WebLogic Web Services do not follow this last part of the specification.</p> <p>WebLogic Server now has a warning message for the data type mapping that is non-compliant with JAX-RPC.</p>
CR217189	<p>Generating a Web Service from a WSDL no longer fails with a compilation error.</p>

<b>Change Request Number</b>	<b>Description</b>
CR217694	Now, <code>weblogic.webservice.core.soap.DetailImpl</code> is serializable.
CR220734	The <code>servicegen</code> ant task no longer fails to process <code>types.xml</code> with schema that has <code>&lt;include&gt;</code> .
CR221530	The ant task and autotype with <code>encoding="soap"</code> no longer results in the following error: Encoding can only be "soap" or "literal".
CR222033	If a user-defined exception starts with the new line character <code>\n</code> , for example, the first character is <code>\n</code> , and the message presented after the <code>&lt;n&gt;</code> character is no longer getting truncated when the exception is thrown.
CR222116	Using <code>anySimpleType</code> extension in <code>complexType</code> no longer causes a <code>weblogic.xml.schema.binding.BindingException</code> in <code>clientgen</code> .
CR222793	When two anonymous types have the same name, running the <code>clientgen</code> Ant Task several times no longer generates different code.
CR222873	When Message Level Security was enabled for Web Services, if the request message exceeded a certain size, WebLogic Server was throwing a <code>JSAFE_InvalidUseException</code> when trying to encrypt the request message.  To address this problem, the system property, <code>-Dweblogic.xml.outputStream.bufferSize</code> , was added.  Use this property in the server start up script to increase the buffer size to accommodate large request messages and allow their encryption. The default value of the buffer size is 1024 bytes.
CR224569	Now, a WebLogic Web Service client can catch a user-defined exception thrown from an AXIS service.
CR225438	When a handler uses <code>SOAPConnection</code> to route a request to a Web Service without any change, if the incoming request has any empty element, the routing handler no longer fails with the following exception:  <code>javax.xml.soap.SOAPException: Failed to send message: java.net.ProtocolException: Did not meet stated content length of OutputStream: you wrote 3653 bytes and I was expecting you to write exactly 3675 bytes!!!</code>

<b>Change Request Number</b>	<b>Description</b>
CR226344	If you throw a user-defined service-specific exception defined as a complex type exception with only one element, WebLogic Server no longer throws the following exception: <code>weblogic.xml.schema.binding.SerializationException</code>
CR227493	Now, the Web Service client works when the input type of the Web Service method is char.
CR232578	WebLogic Server threw a <code>NullPointerException</code> when a message posted by the session bean to the JMS queue was lost while calling an external Web Service.  The message is now redelivered without the <code>NullPointerException</code> .
CR232678	Wsd2service no longer fails to add the parameter <code>location="attachment"</code> to the dynamic wsdl file ( <code>web-services.xml</code> ) when the return param is <code>javax.xml.transform.Source</code> .
CR233130	You can now use the <code>-Dweblogic.webservice.rpc.timeoutsecs</code> system property to set the default rpc timeout value for Web Services.

## WebLogic Tuxedo Connector

Change Request Number	Description
CR121432	<p>In the <code>weblogic.wtc.tbridge.tBexec</code> class, while running <code>tpenqueue</code>, the code checked for null <code>replyQ</code> and replaced it with the name of the configured request queue.</p> <p>This check has been eliminated, and <code>NULL</code> is now a configurable value for <code>replyQ</code>.</p>
CR179410	<p>WTC did not clean up the old connection after <code>reconnect</code>. Tuxedo /Domain and WTC are connected with the <code>ON_STARTUP</code> connection policy. If the machine on which the Tuxedo domain was running was down, when it was restored, WTC did not clean up the old connection.</p> <p>Now, when the same remote domain receives a new connection, the old connection is dropped.</p>
CR196598	<p>WTC memory leak occurs when the Tuxedo side defines unconnected local domain.</p> <p>Now, the memory leak in some connection-termination error scenarios is absent and clean-up logic is implemented.</p>
CR202460	<p>When all <code>ExecuteThreads</code> in WLS are busy waiting for replies in <code>weblogic.iop.SequenceRequestMessage.waitForData</code>, WTC requires at least one more thread to process replies. In this situation, the server is simply blocked forever.</p> <p>Now, the reply is processed by bypassing the thread allocation and directly passed to the <code>IIOP</code> layer.</p>
CR208701	<p>WTC <code>tBridge</code> causes <code>CLOSE_WAIT</code> sockets on Sun Cluster with a firewall between WLS and Tuxedo. WTC <code>tBridge</code> opens a connection to the firewall and the firewall cannot access Tuxedo because Tuxedo is not booted. The WTC connection stays open in <code>CLOSE_WAIT</code> state.</p> <p>The socket connection now closes properly.</p>
CR212218	<p>WTC does not detect network errors between WTC and Tuxedo. WTC waits for the blocking timeout and eventually throws a <code>TPETIME</code> exception.</p> <p>Now WTC behaves like Tuxedo and throws a <code>TPESYSTEM</code> exception to client programs when the connection goes down.</p>
CR213232	<p>WTC Domain level failover fails to work if <code>ConnectionPolicy</code> is set to <code>ON_DEMAND</code> for <code>WTCLocalTuxDom</code> and <code>ON_STARTUP</code> for <code>WTCRemoteTuxDom</code>.</p> <p>Remote connection policy overrides the local connection policy when appropriate.</p>

<b>Change Request Number</b>	<b>Description</b>
CR216682	<p>Unexpected rollbacks were encountered when calling a remote CORBA object from within an EJB with a trans-attribute of RequiresNew.</p> <p>Now, CORBA WTC XA resources are allowed to be suspended and resumed correctly.</p>
CR223488	<p>WTC automatic retry causes thread to hang when network error occurs.</p> <p>Now, when a connection is in progress for one remote domain, it is not included in the list of remote domains available for a given service.</p>
CR228745	<p>When tpcall() is executed from a Tuxedo 8.1 client with the TPNOREPLY flag set, WTC returns a reply despite the flag setting. This problem leads to degraded performance.</p> <p>Now, no reply is sent when the TPNOREPLY flag is passed in inbound requests.</p>
CR236066	<p>WTC unable to map CORBA::UserException over WTC.</p> <p>WTC now injects a reply message with TPSVCFAIL and user data into the RMI subsystem.</p>

## XML

Change Request Number	Description
CR158523	<p>On UNIX, when a private registry is published using uddiexplorer, the following error no longer occurs:</p> <pre>An error has occurred E_fatalError(10500): a serious technical error has occurred while processing the request. 'null[LDAP: error code 65 - Object Class Violation]'</pre>
CR187425	<p>If the MBean Definition File (MDF) name is multibyte characters, WebLogic MBeanMaker no longer throws a <code>java.net.MalformedURLException</code> when generating the MBean type.</p>
CR203364	<p>WebLogic Server no longer throws a <code>NullPointerException</code> in Xerces when using JAXP <code>DocumentBuilder</code> multiple times.</p>
CR205549	<p>JavaServer Faces, a Web Application framework that accomplishes the Model-View-Controller (MVC) paradigm, provides a free implementation called myFaces. When trying to use myFaces, WebLogic Server was throwing the following exception upon startup:</p> <pre>Cannot load Implicit TLDs java.lang.NullPointerException</pre> <p>Now, myFaces is able to load implicit tag library descriptor (TLD) files and the exception is no longer thrown.</p>
CR215016	<p>When doing lookup of DTDs over HTTP from a central Web server, if the URL failed (that is, if the Web server was down momentarily), the XML registry did not renew the DTD in the cache and failed each time the DTD was used.</p> <p>Now, the XML registry renews the DTD in the cache, which resolves the problem.</p>
CR215842	<p>When filtering XML messages using <code>JMS_BEA_SELECT()</code>, WebLogic Server no longer throws an <code>XMLStreamException</code> if the XML fragment in the message includes an attribute.</p>

# Resolved Problems for Service Pack 4

Service Packs are cumulative; Service Pack 4 contains all the fixes made in earlier Service Packs released for WebLogic Server 8.1.

The following sections describe problems that were resolved in WebLogic Server 8.1 Service Pack 4:

- [“Administration Console” on page 4-3](#)
- [“Classloaders” on page 4-8](#)
- [“Clusters” on page 4-8](#)
- [“Connector” on page 4-8](#)
- [“Core WebLogic Server” on page 4-10](#)
- [“Deployment” on page 4-16](#)
- [“EJB” on page 4-20](#)
- [“Installation” on page 4-27](#)
- [“Internationalization” on page 4-27](#)
- [“J2EE” on page 4-27](#)
- [“JCOM” on page 4-29](#)
- [“JDBC” on page 4-29](#)
- [“jDriver” on page 4-32](#)

## Resolved Problems for Service Pack 4

- “JMS” on page 4-32
- “JNDI” on page 4-36
- “JSP” on page 4-36
- “JTA” on page 4-38
- “JTS” on page 4-39
- “JVM” on page 4-39
- “Node Manager” on page 4-40
- “Operations, Administration, and Management” on page 4-41
- “Plug-Ins” on page 4-48
- “RMI/RMI-IIOP” on page 4-54
- “Security” on page 4-57
- “Servlets” on page 4-63
- “SNMP” on page 4-69
- “Tools” on page 4-69
- “Web Services” on page 4-70
- “WebLogic Tuxedo Connector” on page 4-76
- “XML” on page 4-78

# Administration Console

<b>Change Request Number</b>	<b>Description</b>
CR112354	<p>WebLogic Server was checking for the existence of the selected pool name in the ConnectionPool list, but not in the MultiPool list.</p> <p>Now, WebLogic Server checks for the selected pool name in both lists and it is possible to assign DataSource and TXDataSource to a MultiPool in the Admin Console.</p>
CR187204	<p>The performance monitoring graphs in the Admin Console were freezing immediately after garbage collection was forced (the "Force Garbage Collection" button was clicked).</p> <p>Now, the Graph Gather thread does not die until the browser is closed. As a result, the performance monitoring graphs no longer freeze after forced garbage collection.</p>
CR181865	<p>The Admin Console was allowing the addition of members into the JMS distributed destination even if their JMS server was not targeted.</p> <p>Now, a member cannot be added to the JMS distributed destination if the member's JMS server is not targeted.</p>
CR130369	<p>Trying to define a Security Policy on WebService methods that had an array type as a parameter was causing a Javascript runtime error in the Admin Console.</p> <p>Now, WebLogic Server creates the signature of a method with the array parameters in a format that is compatible with the security subsystem. As a result, the Javascript runtime error no longer occurs when the policies for a method containing the array as a parameter are being defined.</p>
CR190752	<p>The RemoveInfectedConnectionsEnabled attribute, a JDBC connection pool specific property, was displayed on the console although none of the other attributes in PoolParamsMBean appeared on the console.</p> <p>Now, the RemoveInfectedConnectionsEnabled attribute is not a displayable attribute, so it is consistent with other attributes in PoolParamsMBean. As a result, WebLogic Server does not throw an exception when the descriptor for Application-Scoped JDBC Connection Pool is edited.</p>
CR190524	<p>After migrating from WebLogic Server 8.1 SP2 to SP3, WebLogic Server was throwing a Console Extension NullPointerException.</p> <p>Now, WebLogic Server has a display name attribute and an alternate name attribute in the catalog which allows the catalog to have more familiar names and maintains compatibility between service packs. As a result, WebLogic Server no longer throws the Console Extension NullPointerException after migrating from WebLogic Server 8.1 SP2 to SP3.</p>

Change Request Number	Description
CR189027	<p>The Console was slow to display the webapp table. The more webapps were deployed, the slower the webapp table displayed.</p> <p>Now, WebLogic Server makes fewer calls to get webapp runtime mbeans. As a result, the webapp table is displayed faster and increases in speed as the number of webapps deployed increases.</p>
CR161951	<p>The Re-order Authentication Providers... link did not appear in the Custom Authentication Providers page when, from the main administration console, clicking the Realms... link, then the myrealm... link, then the Providers Tab, and then Authentication.</p> <p>Now, the Re-order Authentication Provider... link appears when using this navigation in the console.</p>
CR129551	<p>It was not possible to revert to the Default Adjudication Provider from the Custom Adjudication Provider.</p> <p>Code has been added that replaces the Adjudication Provider for the realm even if the provider already existed. As a result, it is now possible to revert to the Default Adjudication Provider.</p>
CR125334	<p>The Servlets table was not sorting arrays properly when "URL Patterns" was clicked. If the attribute was of type array, an exception was being thrown when comparing arrays of different lengths.</p> <p>The compare method for array attributes has been modified so that the attribute of array type is now properly sorted and no exception is thrown.</p>
CR124223	<p>To determine whether a component was targeted to the server, only the server, and not the cluster, was being checked.</p> <p>Now, WebLogic Server checks the server and the cluster to which the server belongs. As a result, the components (application, JDBC connection pool, etc.) that are targeted to a cluster appear as deployed on the server if the server is part of the cluster.</p>
CR111502	<p>Deploying a new WebLogic Portal application via the WebLogic Console while targeting modules did not work with Internet Explorer because the URL size was exceeding its maximum limit.</p> <p>WebLogic Server now passes the names of the EJBs and the webapp in the application as hidden attributes instead of passing them as part of the URL.</p> <p>As a result, an application with any number of components can be deployed without a javascript error.</p>

<b>Change Request Number</b>	<b>Description</b>
CR106773	<p>If an EJB did not have a package, an <code>indexOutOfBounds</code> exception was being thrown when trying to substring the first package name of the full package of the EJB.</p> <p>Now, when the EJB has no package, an asterisk (*), is shown as the default package when defining policies for jCOM component access. As a result, an <code>indexOutOfBounds</code> exception is not thrown if the EJB does not have a package.</p>
CR106457	<p>Only <code>weblogic-rdbms-bean</code> elements of <code>weblogic-rdbms-jar</code> were being converted into XML. As a result, the complete <code>weblogic-rdbms-jar</code> elements were not being shown.</p> <p>WebLogic Server now converts <code>weblogic-rdbms-jar</code> elements into XML so that all elements of the <code>weblogic-rdbms-jar.xml</code> deployment descriptor are displayed.</p>
CR103417	<p>EJB monitoring pages were not refreshing properly because the pages were being cached. The tags that cache the browser pages were removed and now the pages are refreshed properly.</p>
CR093082	<p>The SSL config console page was not letting the user set the Key Alias and Passphrase. The SSL config console page now shows the user set values for Key Alias and Passphrase if the user has changed them.</p>
CR186293	<ol style="list-style-type: none"> <li>1. The left navigation pane did not honor the language set from the browser. It honored the language preferences set from console preferences page.</li> <li>2. The console left and right navigation panes did not honour the language set from the preferences setting page of the console.</li> </ol> <p>The applet was made to pass the <code>accept-language</code> header to server while making any request. WebLogic Server now honors the language set from the browser for the left navigation panel.</p> <p>While the console extension is being deployed it passes the current language to the helper class, which helps it find the correct catalog for the console extension and the extension is correctly localized.</p>
CR178780	<p>In WTC, the imported services, the exported service and the remote domains are considered a tuple where combination of the Resource Name, Local Domain, and Remote Domain must be unique. The Administration Console prevented defining imported services, exported services and remote domains with the same resource name.</p> <p>WebLogic Server now checks for uniqueness of the tuple rather than just the attribute values.</p>

<b>Change Request Number</b>	<b>Description</b>
CR172966	Log messages in the console logs were being truncated.
CR129054	WebLogic Server now processes escape characters correctly, preventing truncation of the log messages.
CR130110	There was an upload directory problem when the complete path rather than a relative path was specified. WebLogic Server now checks for relative and absolute path when uploading.
CR129518	When using a remote machine and trying to upload WAR, RAR, or JAR files, WebLogic Server was unable to find the current path and thus returned the user to the upload screen rather than uploading the files. Weblogic Server is now able to get the current path and upload these types of files.
CR129373	When WebLogic Server passed attributes to html, periods ('.') were being replaced by underscores ('_'). If the qualifying name of an attribute contained an underscore ('_') it was replaced by a period ('.') when it was returned to the server. Because of this, an instance not found exception was thrown WebLogic Server now replaces underscores ('_') in qualifying names with colons (':'). The reverse process occurs during conversion of the attribute name to the qualifying name of the attribute. This way underscores ('_') in the qualifying name are restored.
CR183641	When the Administration Server was used to get information about transactions running on managed servers, the information was incorrect. WebLogic Server now retrieves the transaction information from one of the servers associated with the transaction.
CR184784 CR183444 CR201566 CR183444 CR132832 CR197295	The WebLogic Server Administration Console did not allow certain characters such as spaces and hyphens in the names of groups, users and roles. As a result, user certificates generated by other applications that contained spaces and hyphens could not be validated in WebLogic Server. WebLogic Server now allows user/group/role names to contain these special characters.
CR126074	WebLogic Server used the listBinding method of the Context class, which tried to resolve all objects completely. An exception was thrown if the attempt to resolve the objects failed. Resolving the objects is now done in the Administration Console. If an object can not be resolved, WebLogic Server shows that particular name with red circle.

Change Request Number	Description
CR122403	<p>Filtering was being carried out on the basis of an "if name starts with" criteria. Therefore, it returned servers targeted to other connection pools also, if the name of this pool started with the same characters as the given pool.</p> <p>Filtering has been changed to look for an exact match. So only servers targeted to the given connection pool are returned.</p>
CR122362	<p>The MaxPostTimeSecs attribute was missing from the ServerProtocols general tab in the Administration Console.</p> <p>The ServerProtocols general tab now shows the MaxPostTimeSecs attribute.</p>
CR119472	<p>The setTextId method was not resetting the text, because of which two tags using the same instance were not reflecting the true value.</p> <p>WebLogic Server now resets the text in the setTextId method to show the correct value.</p>

## Classloaders

---

<b>Change Request Number</b>	<b>Description</b>
CR177678	<p>The Class.getSigners() method did not return the certificates if the class was in a signed jar file.</p> <p>Now, WebLogic Server obtains the bytes from the JarFile instead of the ZipFile when loading classes from jar files. As a result, the Class.getSigners() method returns the certificates if the class was loaded from a signed jar file.</p>

---

## Clusters

---

<b>Change Request Number</b>	<b>Description</b>
CR177776 CR201676	<p>WebLogic Server encountered distributed dead-locks in a cluster when replicated session http requests landed on a server that acts as neither primary nor secondary for the requests. If multiple such requests landed on servers in a cluster, all the threads in the default thread pool were being exhausted due to this behavior and at some point in time, there were no threads available in default thread pool to receive responses. This lead into distributed dead-lock.</p> <p>WebLogic Server no longer deadlocks under these conditions.</p>

---

## Connector

---

<b>Change Request Number</b>	<b>Description</b>
CR176209 CR183827	<p>Initially the out debug statement used the transaction object. This caused a ConcurrentModificationException.</p> <p>The transaction object was removed from the debug statement which eliminated the exception.</p>

---

---

<b>Change Request Number</b>	<b>Description</b>
CR190895	<p>Changes were made to significantly improve the performance of the Connector Pool and process connection requests faster.</p> <p>One of these changes is the addition of a new element, use-first-available, to the weblogic-ra.xml descriptor. When use-first-available is set to true the Connector Pool assumes that all connections share the same attributes and that there is no advantage to using one over another for any connection request. Using this option bypasses the need for the Connector Container to call matchManagedConnection on the adapter and thus improves performance.</p>
CR184164	<p>WebLogic Server was throwing a warning (BEA-190085) when a group of RARs were deployed even if the base RAR was deployed first.</p> <p>The log message has been changed to an informational message instead of a warning.</p>

---

## Core WebLogic Server

<b>Change Request Number</b>	<b>Description</b>
CR188709	<p>When the port number was greater than 32K, and the port was read as a short then converted to an int, the result was a negative number because the conversion was not done properly.</p> <p>WebLogic Server now does the conversion properly.</p>
183683	<p>The default identity asserter was not correctly asserting against the default realm.</p> <p>When connecting to a third-party using RMI-IIOP with security enabled, the default identity asserter will now correctly assert against the default realm.</p>
CR197951	<p>When a Managed Server is communicating to an Administration Server via a load balancer with NAT, under certain circumstances, the Managed Server's private IP address, instead of the DNS name, was being used for communication.</p> <p>A utility was written to extract and use the DNS name in cases where both the private IP address and DNS name were provided by the caller. As a result, the server no longer hangs when a Managed Server is communicating to an Administration Server via a load balancer with NAT.</p>
CR190010 CR189462	<p>WebLogic Server was not honoring TransactionalObject interfaces for outbound Interface Definition Language (IDL) calls and not propagating transactions when using these interfaces.</p> <p>The IIOP subsystem now honors IDL interfaces inheriting from TransactionalObject. It is now possible to propagate transactions to Object Transaction Service (OTS) 1.1 ORBs (i.e. IBM ORB on a mainframe) from WebLogic Server.</p>
CR191772	<p>A receiving ORB sending LOCATION_FORWARD in the context of a transaction would cause WebLogic Server to retry the request without a transaction which caused a transaction rollback on the remote side.</p> <p>WebLogic Server now correctly associates a transaction with a request retried as a result of LOCATION_FORWARD. Now, transactional interop with foreign ORBs works correctly.</p>
CR200264	<p>When trying to marshal a CORBA Any containing a CORBA Any, a typecode at position 0 in the stream occurs which results in an assertion error.</p> <p>Now, WebLogic Server accepts a typecode at position 0 in the stream and no assertion error is thrown.</p>

<b>Change Request Number</b>	<b>Description</b>
CR179262	<p>During startup, WebLogic Server was experiencing a deadlock between <code>weblogic.jms.common.DistributedDestinationManager</code> and <code>weblogic.cluster.MemberManager</code>.</p> <p>Synchronization has been reduced in <code>MemberManager</code> and now there is no longer a deadlock.</p>
CR194333	<p>The <code>weblogic.Deployer</code> tool was trying to invoke a <code>doesUserHaveAnyAdminRoles()</code> method on <code>SecurityServiceManager</code> which is not initialized on the client. As a result, it was throwing a <code>NotYetInitializedException</code> if the Admin port was enabled on the server.</p> <p>Now, the <code>weblogic.Deployer</code> tool does not invoke a <code>doesUserHaveAnyAdminRoles()</code> method on the client because <code>SecurityServiceManager</code> is not initialized on the client.</p>
CR191720 CR191721 CR191552	<p>The client was hanging during JNDI authentication if it re-connected after a network partitioning, but before the server recognized the network partitioning.</p> <p>Now, WebLogic Server recognizes that it had missed the <code>PeerGone</code> from a client that did not actually shut down but just had a partitioned network. WebLogic Server then removes the old connection information, creates a new connection, and retries the authentication request. This ensures that the client does not hang.</p>
CR192818	<p>The <code>weblogic.Admin LOCK</code> command is deprecated, but should lock the server, allow only privileged users to connect, and reject any new login attempts. WebLogic Server was allowing new login attempts even after executing the <code>LOCK</code> command.</p> <p>Now, the <code>weblogic.Admin LOCK</code> command works as expected. WebLogic Server only allows privileged users to connect when the server is locked.</p>
CR192644	<p>The interface definition language (IDL) <code>TimeBase</code> was not current. As a result, the <code>org.omg.TimeBase.UtcT</code> class was not compatible with other ORBs.</p> <p>The <code>TimeBase</code> IDL has been updated to match the Object Management Group (OMG) reference IDL for CORBA 2.6. As a result, the <code>org.omg.TimeBase.UtcT</code> class can now be used in any WebLogic Server classloader.</p>
CR192420	<p>When the transaction is propagated from the CORBA Object to the EJB, WebLogic Server exported <code>CoordinatorImpl</code> objects to the rmi tree. These objects were never unexported which caused a memory leak.</p> <p>The mechanics of exporting a <code>CoordinatorImpl</code> object have been changed such that the memory leak no longer occurs.</p>

<b>Change Request Number</b>	<b>Description</b>
CR192362 CR186757 CR200758	<p>Event logging was not working properly in beasvc. Event logging could not locate the event sources and was not reporting the proper message.</p> <p>Now, event logging ensures that the event sources are properly registered when the windows service is installed with beasvc and properly unregistered when the service is removed.</p>
CR180291	<p>If multiple server instances run on the same computer in a domain that uses a domain-wide administration port, a clear error message is displayed that explains what to do. You must either:</p> <p>Host the server instances on a multi-homed machine and assign each server instance a unique listen address, or</p> <p>Override the domain-wide port on all but one of one of the servers instances on the machine. Override the port using the Local Administration Port Override option on the Advanced Attributes portion of the Server-&gt;Connections-&gt;SSL Ports page in the Administration Console.</p>
CR178205 CR121483	<p>Calling getAttributes on the Admin Server's MBean was causing a ConcurrentModificationException.</p> <p>Now, the HashMap is not modified simultaneously by two threads. As a result, WebLogic Server no longer throws the ConcurrentModificationException when getAttributes is called on the Admin Server's MBean.</p>
CR190595	<p>IDL strings over 512k were resulting in a MARSHAL exception when transmitted.</p> <p>The string size limit has been changed to 64M. As a result, WebLogic Server now accepts IDL strings of up to 64M.</p>
CR189462 CR190010	<p>Interface Definition Language (IDL) wide strings were being corrupted when used in collocated IDL invocations.</p> <p>Collocated invocations now use standard IDL stubs rather than WebLogic Server generated stubs. As a result, wide strings are no longer corrupted for collocated calls.</p>
CR189338 CR190012	<p>Distributed garbage collection (DBC) was cleaning up CORBA objects.</p> <p>All CORBA objects are now configured so that they do not use DGC at all but instead to have their lifetime controlled by the ORB.</p>

<b>Change Request Number</b>	<b>Description</b>
CR184182 CR187177 CR199812	<p>WebLogic Server was throwing a NullPointerException when de-serializing the stateful session bean (SFSB) Handle under the following circumstances:</p> <ol style="list-style-type: none"> <li>1) when resolving object in Handle</li> <li>2) when initializing RJVM</li> <li>3) when getting EndPoint in BasicRemoteRef</li> </ol> <p>Now, under these circumstances, WebLogic Server is not throwing any NPEs.</p>
CR190490	<p>The java.net.SocketException: Bad File Number exception was being thrown when the servlet tried writing to a socket.</p> <p>This can happen when the servlet tries writing to a closed socket. The exception is harmless. Code was changed such that this exception is ignored and does not get logged.</p>
CR185841	<p>When starting up many managed servers concurrently, the admin server CPU usage was high. WebLogic Server now makes fewer remote calls to ServerMbean.getName(). As a result, managed server startup time in a large cluster is faster and the CPU usage on the admin server is lower.</p>
CR184487 CR196822	<p>When clients were tunneling http/https over t3, if the stubs were not present in the client classpath, the client was attempting to open a direct connection to the backend WebLogic Server.</p> <p>Now, WebLogic Server ensures that when clients do network classloading, they use the tunnel instead of attempting a direct connection.</p>
CR182838 CR195712	<p>LocalServerRef did not implement the hashCode method which caused multiple entity beans with different PKs to have the same stub.</p> <p>LocalServerRef now correctly implements the hashCode method.</p>
CR175808	<p>The ClusterRuntime MBean name was using the server name instead of the cluster name.</p> <p>Now, the ClusterRuntimeMBean name initialization is correct. The ClusterRuntime MBean name is now correctly reflected in the MBean.</p>
CR175772	<p>WebLogic Server was unable to accept POST requests from a client after the KeepAlive connection was timed out on Linux.</p> <p>WebLogic Server now shuts down the socket's input and output before a socket close. As a result, POST requests from a client after the KeepAlive connection is timed out on Linux are accepted.</p>

Change Request Number	Description
CR126838 CR196450	<p>The WebLogic Server stuck thread warning message was being logged only to the individual server log file as it was just a warning message. It has been changed to an error message and is now logged to the domain level log file as well.</p>
CR188371	<p>When an application cached a stateless session bean remote stub, and all the servers in the cluster were restarted, the stub was unable to refresh its lists of server nodes where the remote object is available and failover did not succeed this was happening because the stub did not have the information needed to re-establish initial context with the cluster nodes, hence the remote method invocation failed.</p> <p>WebLogic Server code was not propagating the thread environment for the stateless session bean stubs in WebLogic Server 6.1 versions and it is required in WebLogic Server 7.0 and higher versions for un-marshalling to set the environment so failover works.</p> <p>The runtime descriptors for the clusterable stateless session beans now have the propagate-environment attribute set to true by default.</p> <p>The descriptor is now read and the propagateEnvironment is set so the environment is passed on during un-marshalling. This allows the failover logic to reconnect to the cluster nodes to retrieve the new list of server nodes where the remote object is available and allow for proper failover.</p>
CR189337	<p>Using CORBA Any's to transmit certain primitive types was corrupting the primitive data contained in the Any. Instead of only masking the non-databits for storage in the Any, some databits were masked as well.</p> <p>The mask was changed and Any's now work as expected.</p>
CR180147	<p>Memory leaks were occurring in the NTSocketMuxer when converting native error messages to unicode.</p> <p>WebLogic Server now releases the character pointer obtained from jni strings, eliminating the memory leak.</p>
CR176331	<p>A NullPointerException was sometimes thrown from the muxer during the cleanup of a socket.</p> <p>This happened when the protocol discriminator did not set the proper muxable socket.</p> <p>WebLogic Server now sets the proper muxable socket during discrimination so that it uses the right muxable socket when delivering exceptions.</p>

---

<b>Change Request Number</b>	<b>Description</b>
CR181986	WebLogic Server running as a service sometimes ran out of memory if it was using a large number of threads.  Reducing the reserve stack size used by beasvc.exe and beasvc64.exe from 1mb to 256 kb eliminated the memory problem.
CR127920 CR190968 CR197708	HeartbeatHelper object, which is used on the client side to ping the server, was being cached and when the server was restarted, the client attempted to ping but the ping was rejected by the new server. This resulted in IllegalStateException. HeartbeatHelper is now looked up from the cos NamingService and is no longer cached.

---

## Deployment

Change Request Number	Description
CR192097	<p>If a Pinned App was deactivated on one server of the cluster, it was being deactivated on all other servers of the cluster wherever it was deployed.</p> <p>The Pinned deployment behavior for the -deactivate, -stop and -unprepare commands have been changed so that now, the Pinned App is only deactivated on the specific clusteredServer where the deactivate command is requested.</p> <p>For example, the command:</p> <pre>java weblogic.Deployer -deactivate -name myPinnedApp -targets MS1</pre> <p>only deactivates the myPinnedApp on MS1 of Cluster CLS1. It will not deactivate the application on other servers of the cluster CLS1.</p> <p>This new behavior will not affect applications that are either deployed to the cluster or pinned to one server in the cluster. It will only affect those applications that are pinned to few servers (more than one) in cluster.</p>
CR177486 CR124375 CR134935 CR173172	<p>WebLogic Server was making too many RMI calls to find out the deployment status of an application/component on a target especially if the target was a cluster or a virtual host. In addition, the module status on the target was not depicting the precise deployment status of the target.</p> <p>Now, the deployment status page is shown in terms of its aggregated deployment status and availability status. The aggregated deployment status of a component could be Available/Not Available for a server and Available/Not Available/Partially Available when it is deployed on a cluster or virtual host. Partially Available implies that the component is available on only some of the servers of the cluster or virtual host.</p> <p>The deployment/availability status is updated when the targeted servers are shutdown either gracefully or forcefully.</p> <p>As a result of all of these changes, WebLogic Server has minimized the number of RMI calls needed to retrieve the status report for a cluster deployment.</p>
CR191908	<p>When using the verbose option on either the Deployer tool or on wldploy ANT task, neither the Deployer tool nor wldploy ANT task was waiting for task completion notifications and they were therefore exiting as soon as they had initiated the task.</p> <p>Now, the Deployer tool and wldploy ANT task wait for task completion notification before exiting even when they use the verbose option.</p>
CR197999	<p>Now, the wldploy ant tool supports the option of deploying staged applications.</p>

Change Request Number	Description
CR197220	<p>The managed server was failing to start up in MSI mode if applications were being deployed using external_stage because the managed server was not able to find the application.</p> <p>Now, the managed server is able to find the application when it is started in MSI mode. As a result, applications are now deployed properly and the managed servers boots properly in MSI mode.</p>
CR125854	<p>Although WebLogic Server does not copy or remove files in no-stage there was no message sent to notify of this.</p> <p>Now, WebLogic Server sends a DeploymentNotification stating that the files cannot be deleted when using the weblogic.Deployer -delete_files option for no-stage applications. The notification is shown only when the Deployer is run in verbose mode.</p>
CR195576	<p>Application Poller was not catching the exception thrown as part of deploying any application with an invalid directory structure from the applications folder. As a result, the remaining applications failed to be deployed after any such invalid application was deployed.</p> <p>Now, Application Poller catches exceptions thrown while deploying any applications with an invalid directory structure from the applications folder so that the invalid applications failure is logged and the remaining valid applications are deployed successfully.</p>
CR192094 CR195074	<p>There was a memory leak in the way WebLogic Server was using deployment task objects for historical purposes. WebLogic Server was running the OutOfMemoryError after a certain number of deployment actions (deploy, undeploy, stop, start, redeploy, etc.). For large applications' deployment actions, WebLogic Server was running the OutOfMemoryError quickly.</p> <p>Changes have been made to the code so that it will not leak memory due to these deployment task objects.</p>
CR193897	<p>WebLogic Server was not automatically redeploying a WebApp if the Servlet Filter implementation was in the system classpath and the servlet class was modified.</p> <p>Now, the Servlet Filter considers the changes in WebApp classes even though the filter is loaded by the system classloader.</p> <p>When the servlet-reload-check-secs flag is set to something other than -1, this feature is best used in development mode and not in production mode.</p>

Change Request Number	Description
CR192196	<p>A WebLogic Server 8.1 client calling <code>DeployerRuntime.getDeployerRuntime(MBeanHome)</code> was getting an <code>InstanceNotFoundException</code> if invoked against a WebLogic 7.0 admin server.</p> <p>A change was made to the implementation of <code>DeployerRuntime.getDeployerRuntime(MBeanHome)</code>. As a result, a WebLogic Server 8.1 client is now able to fetch the <code>DeployerRuntime</code> MBean from the WebLogic 7.0 admin server and vice versa without getting any <code>InstanceNotFoundException</code>.</p>
CR128977	<p>The <code>weblogic.Deployer</code> utility was not deploying to default targets as it should.</p> <p>Default targets have been set to already deployed targets and not to the Admin Server when deploying or starting applications using <code>weblogic.Deployer</code>. As a result, <code>weblogic.Deployer</code> now deploys to default targets as expected.</p>
CR190088	<p>When only some modules of the application were deployed to a server during redeployment, the application lifecycle listener <code>preStop</code> event was not fired.</p> <p>Now, WebLogic Server calls application lifecycle listener <code>preStop</code>, as part of redeployment, when there is a split deployment (some modules deployed on one target and the remaining modules deployed on a different target).</p>
CR189712 CR175585	<p>The status of certain deployment tasks like <code>deploy/redeploy/distribute/remove</code> was shown as "Failed" in the WebLogic Server admin console when the deployment task targets were not up or reachable. These tasks were actually deferred and not failing.</p> <p>Now, the status of such tasks are shown as "Deferred" in the console.</p> <p>When such tasks are actually completed on the target, the deployment/availability status (module status) of the app/component in question is appropriately updated. However, the deferred status shown in the "Status of last action" column in the console is not updated. The last action status does not represent the current deployment status of the app/component.</p> <p>Also, when tasks are deferred, WebLogic Server now throws the <code>weblogic.management.DeferredDeploymentException</code> instead of the <code>weblogic.management.ManagementException</code>.</p>

Change Request Number	Description
CR092875	<p data-bbox="357 418 1150 470">Deployer was setting the source option to null for redeploy. This resulted in upload not working for redeploy because upload needed source option.</p> <p data-bbox="357 487 1150 569">Deployer now allows redeploy with upload which allows application files to be added, modified, or deleted on the target. These changes are also applied to the application's uploaded source on the Admin Server.</p> <p data-bbox="357 586 1150 670">Upload option has been deprecated. A new option sourcerootforupload has been introduced that allows the user to specify the source root directory or archived application jar file on the remote machine. This file is then uploaded to the Admin server upload directory.</p>
CR186042	<p data-bbox="357 699 1193 751">Static files were not redeployed on managed servers if the managed servers were down at the time of static file redeployment.</p> <p data-bbox="357 769 1193 821">When the managed server comes up, it restages the entire application, allowing the modified static files to be redeployed.</p>
CR184089	<p data-bbox="357 850 1193 932">Restarting the Administration Server reset the drs version when discover was set to false. This caused the Administration Server to be on a lower version thus redeployment of an existing application onto the cluster never succeeded.</p> <p data-bbox="357 949 1193 1003">The Administration Server now knows its drs version after restarting the admin server when discover is set to false.</p>
CR127141	<p data-bbox="357 1032 1193 1053">During a graceful shutdown of the server, applications were undeployed in an arbitrary order.</p> <p data-bbox="357 1071 1193 1185">WebLogic Server now un-deploys applications in the exact reverse order of the deployment. That is, the application with highest LoadOrder number is undeployed first then the next high LoadOrder application will be undeployed etc. Applications with the same LoadOrder will be deployed in alphabetical and undeployed in reverse alphabetical orders.</p>
CR110687	<p data-bbox="357 1215 1159 1267">An application's removal and redeployment logic did not work properly when a targeted Managed server was shut down while a new EAR was being deployed.</p> <p data-bbox="357 1284 1159 1333">The logic now works correctly when a targeted Managed server is shut down during deployment.</p>

## EJB

Change Request Number	Description
CR197639	CMP20 EJBs in WebLogic Server now support automatic primary key generation for DB2 and Informix databases.
CR186325	<p>In WebLogic Server 8.1, compliance check was added so that key-cache-size had to be equal to the increment value on the sequence in the database. If the key-cache-size was not equal to the database increment value, WebLogic Server threw the following error during deployment:</p> <p>The ORACLE SEQUENCE named 'test_sequence' with INCREMENT value '1' '[EJB:011065]The ORACLE SEQUENCE named 'test_sequence' does not have INCREMENT value '1' in the database'</p> <p>Now, WebLogic Server allows the key-cache-size to be less than or equal to the increment value of the sequence in the database. An exception is thrown only if the key-cache-size value is greater than the increment value in the database because that may result in duplicate keys being generated.</p> <p>The following rules are followed while verifying the sequence during the deployment of the EJB:</p> <ol style="list-style-type: none"> <li>1) Allow key-cache-size &lt; DB INCREMENT_BY with a warning that missed keys (10, 20, 30) would be generated.</li> <li>2) Allow key-cache-size = DB INCREMENT_BY. This is as it works now.</li> <li>3) Disallow key-cache-size &gt; DB INCREMENT_BY and disallow the deployment as it could generate duplicate keys.</li> </ol> <p>Other rules regarding the different values of &lt;create-default-dbms-tables&gt; have been modified accordingly.</p> <p>In general, Prod sequences (i.e., sequences defined by the user) are not altered. Dev sequences (i.e., sequences created by Web Logic Server) are altered.</p> <p>If a sequence generation in the order of 1, 5, 10 ... or negative sequence generation -1, -5, -10 ... is required, set the key-cache-size to 1 with INCREMENT_BY in the database to 5. However, because the key-cache-size is 1, the container always goes to the database to get the next autoKey and thus follows the INCREMENT_BY set in the database sequence.</p>
CR184780 CR128984	<p>An Oracle SELECT_HINT that was defined for a finder in a weblogic-ql element was not being included in the generated SQL query.</p> <p>Now, it is included.</p>

<b>Change Request Number</b>	<b>Description</b>
CR198962 CR201394	<p>The EJB Compiler was giving errors when compiling EJBs that had ejbSelect queries that used IS EMPTY or IS NOT EMPTY in the WHERE clause.</p> <p>The ejbSelect queries no longer cause errors under these conditions.</p>
CR193685	<p>Container-managed persistence (CMP) now uses PreparedStatement to execute SQL.</p>
CR191351	<p>Automatic Key Generation was generating the new key from the default number in the database plus the cache size which is defined by &lt;key-cache-size&gt;.</p> <p>Now, WebLogic Server has added the select-first-sequence-key-before-update tag in the auto-key-generation. This tag is used to specify the behavior of automatic primary key generation.</p> <p>If the select-first-sequence-key-before-update is set to true, and when the EJB container needs to fetch the sequence value from a database, it generates the primary key from the current value plus one in the sequence table and updates the value in the table with the current value plus the key-cache-size. For example, when the current value in the sequence table is 100 and the key-cache-size is 100, the EJB container generates the primary key with 101 (the current value plus one in the table), and updates the value in the table with 200 (the current value plus the key-cache-size).</p> <p>By default, the select-first-sequence-key-before-update is set to false.</p>
CR192682 CR106774	<p>When using EJB CMPs, WebLogic Server was translating the EJB-QL keyword, between, to &gt;=? AND &lt;=? . As a result, queries (especially on large tables) with DB2 on some platforms were taking a long time.</p> <p>Now, the queries on DB2 run faster because they use the EJB-QL keyword, between, instead of, &gt;=? AND &lt;=? .</p>
CR183557	<p>CMR collection contained two relationship entities instead of one when the parent and child were created in one transaction. This manifested only when the bean used auto-key-generation for SqlServer and delay-database-insert-until was set to ejbCreate.</p> <p>Changes were made to correct the issue.</p>
CR189847	<p>Stateful session bean (SFSB) InMemory replication was causing an apparent memory leak. Although no memory leak was occurring, the heap memory was not being used efficiently. Now, during SFSB InMemory replication, the heap memory is being used more efficiently.</p>

Change Request Number	Description
CR191003 CR186958	<p>Validation with MetaData was failing while deploying an EJB when the fk-column of a bean also mapped to a cmp-field in the same bean.</p> <p>Now, when the foreign-key column of a bean also maps to a cmp-field in the same bean WebLogic Server does not add the fk-column if it has already been added. As a result, validating the column names when using MetaData no longer fails.</p>
CR190831	<p>When an array of inner class was being used for the EJB method return type, the incorrect code was being generated.</p> <p>Now, WebLogic Server converts the inner class representation and generates the correct code when handling an array of inner class.</p>
CR177114	<p>WebLogic Server was throwing an exception when verifying the sequence during EJB deployment if:</p> <ul style="list-style-type: none"> <li>1) a global synonym was being used and no schemaName was specified or</li> <li>2) the sequence was owned by a user other than the user who created the connection pool for the EJB.</li> </ul> <p>Now, WebLogic Server verifies the Oracle sequence against the ALL_SEQUENCES view even if the schema name has not been specified. As a result, WebLogic Server no longer throws the verification exceptions during EJB deployment.</p>
CR173260	<p>WebLogic Server was throwing an OptimisticConcurrencyException when order-database-operations was set to true, and if an entity bean was removed and created with the same Primary Key in the same transaction.</p> <p>Now, WebLogic Server does not throw the OptimisticConcurrencyException under these conditions.</p>
CR135125	<p>When a setter method for a read-only EJB was called, the setter was not executed but no exception was being thrown.</p> <p>Now, if a setter for a read-only EJB is called, WebLogic Server throws the EJBException.</p>

Change Request Number	Description
CR131561	<p data-bbox="357 418 1193 499">TableQuery verification for a CMP EJB was causing a full table scan on the DB2 database that was running on the host because the WHERE-Clause of the verification query was not being optimized by the DB2 query optimizer. This resulted in long query execution time.</p> <p data-bbox="357 517 1193 598">Now, the "FETCH FIRST 1 ROWS ONLY" option instead of the "WHERE 1=0" option is used for DB2 database-type in the case of TableQuery validation. As a result, performance is now optimized.</p> <p data-bbox="357 616 1193 670">This change is specific to the DB2 database type and is not applicable to any other database type. Other database types still use "WHERE 1=0" in the case of TableQuery validation.</p>
CR128850	<p data-bbox="357 699 1193 781">Even after setting enable-call-by-reference to false, a method parameter was being passed by reference. This behavior is correct. The parameter was being passed by reference instead of by value because it was not serializable.</p> <p data-bbox="357 798 1193 881">Now, if enable-call-by-reference is set to false and business methods of the remote interface have argument types that are not serializable, WebLogic Server shows a warning to explain that the parameter will be passed by reference.</p>
CR187121	<p data-bbox="357 911 1193 1025">A high value for idleTimeoutSecs, for instance, 60000000, in the Deployment Descriptor when multiplied by 1000 to convert it into msec was overflowing into a negative value. This caused the trigger that cleans the passivated beans from the disk to constantly fire, causing high CPU usage.</p> <p data-bbox="357 1043 1193 1124">The variables within the EJB container which held the timeout values in milliseconds, such as idleTimeoutMS, sessionTimeoutMS, and readTimeoutMS, have been changed from the int type to long. This prevents any numeric overflow.</p>
CR122203	<p data-bbox="357 1154 1193 1263">The EJB specification requires cmp and cmr fields to start with lower case characters. Some charsets, however, do not have a notion of upper or lower case. If such a charset was used, our compliance checker would throw an exception because the cmp\cmr fields didn't start with lower case.</p> <p data-bbox="357 1281 1193 1333">Now, the compliance checker only throws the compliance exception if the cmp\cmr field starts with an upper case letter and not if the charset has no notion of upper or lower case.</p>
CR189544	<p data-bbox="357 1362 1193 1416">When an inner class within an inner class was passed as an argument to an EJB method, the incorrect code was being generated.</p> <p data-bbox="357 1433 1193 1489">WebLogic Server now converts the inner class representation and handles an inner class within an inner class correctly.</p>

<b>Change Request Number</b>	<b>Description</b>
CR087261	<p>The EJBDeployer was writing an incorrect deployment message to the log for Message Driven Beans.</p> <p>The correct message is now being logged when a Message Driven Bean is deployed.</p>
CR055396	<p>When a EJB QL syntax error occurred, WebLogic Server generated an error message with an incorrect xml file reference.</p> <p>WebLogic Server now generates the message as follows if there are syntax errors in EJB QL.</p> <p>[java] ERROR: Error from ejbc: Error while reading 'META-INF/ejb-jar.xml' or 'META-INF/weblogic-cmp-rdbms-jar.xml'.</p>
CR188814	<p>PreparedStatement and ResultSet were not closed after updating a BLOB or CLOB column.</p> <p>WebLogic Server now closes PreparedStatement and ResultSet when updating a BLOB/CLOB.</p>
CR188022	<p>In certain conditions while inserting rows at commit time, the EJB container threw a DuplicateKeyException reporting an incorrect primary key as the cause.</p> <p>The exception handling was simplified while inserting rows during commit time. When inserting a Batch of statements, if there is a SQLException, then it is thrown as is. The SQLException will have the details of the failure. This is because of driver limitations around reporting the exact statement within the batch that caused the failure. If the row is not inserted as a part of the Batch, only then a check will be done to detect a Duplicate key and report it.</p>
CR187304	<p>When using Database concurrency strategy, if a BMP tried to create a duplicate row in the same transaction, a ClassCastException occurred.</p> <p>When creating beans, WebLogic Server now checks if the bean already exists.</p>
CR186949	<p>The EJB container was starting the JMS connection before creating the initial number of beans in the free pool. This sometimes caused the container to instantiate another bean instance to process the message delivered to the container.</p> <p>WebLogic Server now creates the configured number of initial-beans-in-free-pool instances before starting the JMS Connection to accept messages.</p>
CR178404 CR192516	<p>If the NOT logical operator was used in a WHERE clause and its operand was a conditional expression which included an OR operator (for example, WHERE NOT (lefthand_cond_exp OR righthand_cond_exp)), a NullPointerException occurred while parsing a query.</p> <p>WebLogic Server now handles the OR operator properly.</p>

Change Request Number	Description
CR175158	<p>BMP Entity beans using the Exclusive concurrency strategy and loaded into the cache did not timeout upon idle-timeout-secs expiration. The same BMP beans did timeout for other types of concurrency strategy such as database, readonly and optimistic.</p> <p>BMP beans using the Exclusive concurrency strategy are now scrubbed from the cache after being idle for idle-timeout-seconds.</p>
CR173231 CR189040	<p>When an inner class was passed as an argument to an EJB method, such as <code>ejbCreate</code>, the generated code did not convert the inner class argument to a correct representation.</p> <p>WebLogic Server has now fixed code generation so that it converts the inner class argument to a correct inner class representation.</p>
CR127369 CR195223	<p>An <code>AssertionError</code> was sometimes thrown when more than one bean was based on the same Java class.</p> <p>This error occurred when the following conditions were satisfied:</p> <ol style="list-style-type: none"> <li>1. a bean A had a many-to-one relationship to a bean X (unidirectional relationship)</li> <li>2. a bean B had a many-to-one relationship to a bean X (unidirectional relationship)</li> <li>3. beans A and B were two different deployments based on the same java class.</li> </ol> <p>While processing the relationships of beans, WebLogic Server holds the list of <code>cmr-field</code> names, and if the <code>cmr-field</code> name has not been declared, WebLogic Server creates it based on the bean class name. In the above case, while processing relationships of bean X, the <code>cmr-field</code> names of the relation to bean A and the one to bean B will be created. But these class names are the same, so the created <code>cmr-field</code> names are the same. This causes the <code>AssertionError</code>.</p> <p>Code has been added to make the <code>cmr-field</code> name unique, eliminating the possibility of conflicting names.</p>
CR184154	<p>The Weblogic Server generated java class for <code>weblogic-cmp-rdbms-jar.xml</code> during <code>ejbc</code> released the Resultset incorrectly. This caused the JDBC log to overflow with "java.sql.SQLException".</p> <p>Code was added which corrected the order in which the JDBC objects are closed in the EJB generated code.</p> <p>WebLogic Server now releases the correct Resultset.</p>

---

<b>Change Request Number</b>	<b>Description</b>
CR132853	<p>When a Message Driven Bean uses the synchronous message polling scheme, it has issues when the Sonic.jms server is used. The message driven bean container's polling optimizations could result in a delay in the message receiving.</p> <p>To avoid this problem, do not use the optimized poller. As the Sonic message delivery scheme does not work well with this scheme, use a poller that continuously polls the Sonic.jms server.</p> <p>This new Message Driven Bean behavior is applicable to TIBCO and Sonic.jms providers only</p> <p>See also CR128980, CR135722</p>

---

## Installation

---

<b>Change Request Number</b>	<b>Description</b>
CR121040	<p>It was not possible to change the password of a WebLogic Server Windows Service without having to uninstall and re-install the service.</p> <p>WebLogic Server now has an edit option so that parameters can be edited without having to re-install the service.</p> <p>Individual or multiple options can be edited simultaneously as follows:</p> <pre>beasvc -edit -svcname:wls_service -javahome:"C:\java\java142_05" -password:"blah" (this command alters the value of "javahome" and "password" parameters)</pre> <p>The options will take effect when the service is restarted.</p>

---

## Internationalization

---

<b>Change Request Number</b>	<b>Description</b>
CR079432	<p>MessageLocalizer was not setting the <code>l10n_package</code> attribute in localized catalog files using the <code>l10ngen</code> utility.</p> <p>MessageLocalizer now correctly sets the <code>l10n_package</code> attribute in a localized catalog file.</p>

---

## J2EE

---

<b>Change Request Number</b>	<b>Description</b>
CR192609	<p>When the <code>classloader-structure</code> feature was used in <code>weblogic-application.xml</code>, the <code>ClasspathServlet</code> was unable to serve the files requested in a form such as:</p> <pre>http://host:port/bea_wls_internal/classes/appName@compName/mydir/Foo.class</pre> <p>The <code>ClasspathServlet</code> code has been modified to address this specific requirement.</p>

---

---

<b>Change Request Number</b>	<b>Description</b>
CR125308	Updates to the weblogic-application.xml file were not getting loading upon redeployment. Now, the JDBCModule considers changes to the JDBC connection pool definition in the weblogic-application.xml file upon deployment.
CR103309	Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04-70.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04-70.00.jsp</a> .
CR195836	
CR197966	
CR132360	The weblogic.j2eeclient.Main did not work in webstart. The client jar file is an argument loaded by the webstart client and made available in the classpath. But the file was not available in the current directory resulting in an error.  A system property (weblogic.j2ee.client.isWebStart) was added to WebLogic Server to load the client jar file from the system classpath. weblogic.j2eeclient.Main now works in webstart.

---

## JCOM

---

<b>Change Request Number</b>	<b>Description</b>
CR184188	JCOM_OUTGOING_CONNECTION_TIMEOUT was not being honored when using java2com.
CR187469	WebLogic Server now sets the SO_TIMEOUT to the value specified in
CR202505	JCOM_OUTGOING_CONNECTION_TIMEOUT.
CR198780	When WebLogic Server is started with this system property, com calls are interrupted after the timeout.

---

## JDBC

---

<b>Change Request Number</b>	<b>Description</b>
CR179600	<p>Under certain statement failure conditions, the pool's statement cache would neglect to close statements which led to a cursor leak in the DBMS session.</p> <p>Changes were made to resolve this problem. The cursor leaks no longer occur.</p>
CR190700	<p>Calls to JDBC setTypeMap(map) and getTypeMap() were failing on XA Connection(JTA) with the following exception:</p> <p>java.sql.SQLException: java.sql.SQLException: Cannot set type map</p> <p>The setTypeMap(map) and getTypeMap() APIs have been implemented in JTAConnection. Now, when calls are made to these APIs, there is no longer an SQLException.</p>
CR183884	<p>When creating a XA JDBC connection pool using the Informix JDBC Driver, the hostname and servername properties were being transposed in the generated URL.</p> <p>Now, the hostname and servername properties are in the correct place in the generated URL.</p>
CR200672	<p>The suggestion for MS and Sybase pool's test SQL was "select count(*) from sysobjects". Because this is a huge system table, the query was very slow and problematic for any DBMS concurrency that involved DDL, etc.</p> <p>The suggested/default query text has been changed. Now, the query is no longer slow and it causes no problems for any DBMS concurrency.</p>

---

Change Request Number	Description
CR200403	<p>To get traces when calling the <code>resetStatementProfile()</code> method, WebLogic Server had to be restarted even if tracing was re-enabled.</p> <p>Now, WebLogic Server ensures that the existing file and writer are both closed and never re-used before trying to create new ones. As a result, it is now possible to get traces when calling the <code>resetStatementProfile()</code> method without restarting WebLogic Server.</p>
CR199348	<p>An external client that was using RMI connections to pools and closing them properly, was still getting warnings about a leaked connection from the <code>finalize()</code> method of the RMI <code>SerialConnection</code> object.</p> <p>Changes were made to the <code>is-closed</code> flag so that the client no longer receives these incorrect warnings.</p>
CR197163	<p>The JDBC connection pool's testing of connections was consuming a lot of database resources because each test was creating a new plain statement which requires the DBMS to parse and plan the test SQL every time.</p> <p>Now, pools will reuse a single prepared statement for a connection's test which results in improved performance. However, if any application DBMS tables or procedures are referred to in the test SQL and if they structurally change at runtime, such as an index being added, this may invalidate the test <code>PreparedStatement</code>'s query plan. As a result, the subsequent test will fail and the connection and test statement will be replaced. The test SQL suggested by the Admin Console will typically not include any structurally changing table, so the problem of needlessly recycling a connection is now minimized.</p>
CR196738	<p>The <code>MSSQLServer4</code> driver's <code>Statement.cancel()</code> method was corrupting the JDBC connection. JDBC statements were not being closed within a transaction after the connection was closed.</p> <p>The <code>MSSQLServer4</code> driver's <code>Statement.cancel()</code> method was changed so that it is now a no-op and therefore no longer corrupts the connection. This change has no effect on rolling back transactions.</p>
CR194051	<p>WebLogic Server was leaking Oracle Database cursors when <code>MetaData.getColumns()</code> was called inside a transaction and <code>ResultSet</code> had not been explicitly closed.</p> <p>Now, when the connection is closed, all created <code>PreparedStatements</code> and <code>ResultSets</code> are also closed. As a result, calls to <code>MetaData.getColumns()</code> no longer leak Oracle Database cursors.</p>

<b>Change Request Number</b>	<b>Description</b>
CR189281 CR191429	<p>WebLogic Server was neglecting to retain a stack trace where the connections were made, so when/if the connections leaked, the message contained no helpful information.</p> <p>Now, WebLogic Server delivers a stack trace of where the connections were created which can be useful in diagnosing code that later neglected to close the connection.</p>
CR183355	<p>WebLogic Server was not returning the JTS connection to the pool when the initialization for the connection was failing.</p> <p>Now, if the connection initialization fails, WebLogic Server returns the pool connection.</p>
CR175793	<p>When LogRemoteExceptionsEnabled during server startup, an InstanceNotFoundException was being thrown.</p> <p>Now, instead of calling AdminMbeanHome, WebLogic Server uses the local MBeanHome to getMBean and no exception is thrown when using LogRemoteExceptionsEnabled during server startup.</p>
CR186957 CR198022	<p>When a result set was closed more than once SQLExceptions appeared in the jdbc.log file.</p> <p>WebLogic Server now ignores repeat closes, eliminating the SQLExceptions in the jdbc.log file.</p>
CR182879	<p>WebLogic Server did not throw an error during startup when a MultiPool and a ConnectionPool had the same name, and a DataSource/TxDataSource was targeted at the name.</p> <p>WebLogic Server now throws an error under these conditions.</p>
CR181368	<p>WebLogic Server assumed that, during cleanup, when it found a connection in an autoCommit(false) mode that resetting it back to autoCommit(true) mode would release any DBMS locks. This is untrue for Oracle.</p> <p>WebLogic Server rolled back any transactional context as a part of interrupting any transaction, but the idle-connection code had no knowledge of whether there was a transaction underway, and the cleanup() call it made, was not sufficient for a user-constructed local Oracle transaction.</p> <p>inactive-conn-timeout now rolls back any transaction context during cleanup().</p>
CR184819	<p>WebLogic Server multipools were not failing over when multiple instances of RAC were configured and the first RAC instance was stopped.</p> <p>Multipool now fails over correctly.</p>

## jDriver

Change Request Number	Description
CR185527	<p>When using the JDriver for XA, accessing metadata using DataBaseMetadata sometimes failed and resulted in a java.sql.SQLException.</p> <p>Now, the DatabaseMetaData object retains the reference to the statement it uses for obtaining metadata information, so it is not garbage collected quickly. As a result, accessing the metadata no longer fails.</p> <p>For Database Management Systems that retain cursors for open statements, the retained statement will demand one DBMS cursor until the DatabaseMetaData object is garbage collected.</p>

## JMS

Change Request Number	Description
CR173542 CR188526	<p>A Java-level deadlock was occurring among several cluster members when using JMS distributed queues. The locked objects were weblogic.jms.backend.BEQueue and weblogic.jms.common.DistributedDestinationManager.</p> <p>This problem was resolved by reworking the object synchronization.</p>
CR197857	<p>JMS was not sustaining the minimum flow rate. The actual minimum flow rate was approximately double the configured value. When using the default flow control settings on a Connection Factory, FlowMinimum=50, JMS was sustaining a flow minimum of approximately 100 messages per second rather than 50 messages per second.</p> <p>Now, once a JMS destination exceeds its specified bytes or messages threshold and flow control is started for the producer, flow control throttles back until it reaches FlowMinimum (default 50) and it should maintain that minimum flow rate as long as thresholds are exceeded.</p> <p>This change causes a greater slow down from flow control than previously seen because JMS is now correctly sustaining the minimum flow rate.</p> <p>To maintain previous flow control behavior (behavior prior to the release of 8.1 sp4), double any FlowMinimum configurations and add a FlowMinimum=100 on ConnectionFactories that were previously assuming default values.</p>

<b>Change Request Number</b>	<b>Description</b>
CR192455	Due to an enhancement, the performance of the JMS JDBC Store is now significantly improved. As a result, the JDBC store will now reserve two JDBC connections from the configured connection pool rather than one.
CR190843	Using views on JMSState and JMSStore tables along with Oracle 10G driver and 8i server resulted in SQLExceptions when the JMSServer was booted.  Now, JMSServer boots without error when the customer is using a view for JMSStore and JMSState tables along with Oracle 10G driver and Oracle 8i server.
CR190438	A Java-level deadlock was found between objects weblogic.jms.backend.BEQueue and weblogic.jms.backend.BEConsumer.  This problem has been resolved by updating the lock hierarchy to make sure that the destination has been locked down first.
CR192466	The Administration console was updated to provide a simplified process for configuring a messaging bridge.
CR135578 CR200466	The bytes pending counter for non-durable topics was not getting updated correctly when the consumer exited with outstanding non-durable topic messages still pending Redelivery delay on the topic.  Now, the bytes pending counters for JMS statistics will be appropriately adjusted for non-durable topic messages when the consumer exits and Redelivery parameters have been configured. There is no redelivery attempt on unacknowledged non-durable topic messages when the consumer goes away.
CR189899	For synchronous API calls, when a peerGoneException was detected, the exception was being directed to the ExceptionListener. This behavior violated the spec for synchronous API calls.  Now, JMS only delivers exceptions to the ExceptionListener that have no other place to be reported. PeerGoneExceptions are not delivered to the ExceptionListener if the exception is detected during a synchronous JMS API call. In those cases, the exception is thrown from the synchronous JMS API as a LostServerException.
CR185785 CR126613	When a transaction was rolled back by the transaction manager during onMessage() it was not getting properly cleaned off the thread.  Now WebLogic Server no longer throws AssertionErrors during MDB message processing when a transaction is rolled back during onMessage() processing due to a transaction TimedOutException.

<b>Change Request Number</b>	<b>Description</b>
CR184045	<p>JMS was logging an exception error when it failed to un-register a migratable JMSServer on shutdown.</p> <p>Now, JMS no longer logs an exception error when it fails to un-register a migratable JMSServer that is already shutdown or suspended.</p>
CR189957 CR191998 CR195727	<p>When a disconnect and reconnect occurred with a remote thin client connection before the JMS Server detected peerGone on the initial client connection, JMS was incorrectly cleaning up both the old and the new connections when it detected the lost connection.</p> <p>WebLogic Server now behaves as expected and no longer loses the new connection.</p>
CR188264 CR194959 CR188391 CR190095	<p>JMS was not properly managing its window pipeline (MessagesMaximum) when redelivery limits were reached.</p> <p>WebLogic Server now properly adjusts the window pipeline when redelivery limit is reached to prevent a stall in message processing.</p>
CR187945	<p>JMS DistributedDestination load balancer was not recognizing pre-existing consumer(s) when a member was added dynamically.</p> <p>WebLogic Server now checks to see if a member has any consumers when a member is added to a distributed destination so that the load balancer has the information it needs to load balance correctly.</p>
CR188040	<p>WebLogic JMS was not receiving notification when ServerDebugMBean() JMS attributes were changed.</p> <p>The ServerDebugMBean() can now be used to dynamically enable and disable "DebugJMS" ServerDebugMBean attributes. This allows customers to dynamically enable or disable JMS Debugging flags.</p>
CR184821	<p>In MessagingBridge.ProcessMessage(), if WebLogic Server did not receive any message, it rolled back the transaction and returned from the method. Retry happened with scan_unit delay, which introduced a one second delay.</p> <p>Now, in MessagingBridge.ProcessMessage(), if WebLogic Server does not receive a message, it rolls back the transaction and continues and retries again immediately instead of waiting for scan_unit to kick off a retry after one second. This change has avoided a latency delay in the bridge.</p>

Change Request Number	Description
CR180414	<p>When JMSEnumeration provided a message to the client, it did not make a copy of the message. Anyone using a browser inside the server (such as a servlet) was able to edit the messages that went to others.</p> <p>WebLogic Server now provides a copy when the browser is using a local connection.</p>
CR182338	<p>A receiver stops processing messages when a RedeliveryLimit is configured and the number of times the RedeliveryLimit is reached is greater than the MessagesMaximum setting on the ConnectionFactory.</p> <p>Example: RedeliveryLimit=0, MessagesMaximum=10. Receiver receives a message and calls session.recover() 10 times. Receiver will stop processing messages and the console will show 10 messages pending.</p> <p>Code was added to adjust the window counter when a message is removed from the queue because the RedeliveryLimit had been reached.</p>

## JNDI

<b>Change Request Number</b>	<b>Description</b>
CR191092	<p>When a managed server hosting a distributed topic did a forced shutdown and restarted, the distributed topic never received messages even after the restart.</p> <p>Code was changed in the clustering layer so that the topic is no longer prevented from being re-added to the distributed topics list.</p>
CR186048	<p>While deploying a war file, WebLogic Server creates an environment for the Web Application. To build this environment WebLogic Server creates an InitialContext to look up objects from the JNDI tree. Since jndi.properties was in the classpath WebLogic Server created the InitialContext from the properties specified in jndi.properties.</p> <p>WebLogic Server never closed this initialcontext even after the environment was built. so the user specified in jndi.properties remained on top of the stack which caused problems when the server tried to change the MBean attributes as part of deployment.</p> <p>WebLogic Server now closes the InitialContext after the environment has been built.</p>

## JSP

<b>Change Request Number</b>	<b>Description</b>
CR185870 CR197362	<p>Using wl:cache tags in JSP files was taking a long time and causing a deadlock when running multiple instances of the same JSP file.</p> <p>Weblogic Server JSP container code was corrected to avoid the deadlock situation. As a result, using wl:cache tags in JSP files no longer causes a deadlock.</p>
CR185726	<p>Running weblogic.jspc with over 2000 JSP files caused an OutOfMemory error.</p> <p>The flag, -Dweblogic.jspc.useUniqueJspClassLoader, was added so that when it is set to true, each JSP is loaded in its own classloader, thus avoiding the OutOfMemory Exception.</p>
CR197629	<p>The async attribute of the wl:cache tag was not working because it was deprecated.</p> <p>The async attribute was added back. Now, the wl:cache tag works correctly. The JSP page cannot flush data before wl:cache tag is used if async="true" is set.</p>

Change Request Number	Description
CR136784	<p>The weblogic.Deployer utility could not be used to redeploy a partial web application if pageCheckSeconds was set to -1 in the weblogic.xml file for the web application.</p> <p>Now, weblogic.Deployer can redeploy partial modified web applications even if pageCheckSeconds is set to -1 in the weblogic.xml file.</p>
CR191143	<p>The text in the body of an XML element that represented neither a standard action nor a custom action was not being translated.</p> <p>Now, WebLogic Server passes this text to the current value of out and translates it.</p>
CR131694	<p>On AIX, ISO-2022-JP characters were sometimes garbled because an escape sequence was missing.</p> <p>On AIX, ISO-2022-JP characters are no longer garbled.</p>
CR180854	<p>When a JSP parsing error happened, the JSP source file was locked because the file streams were not closed. Thus the JSP file could not be updated.</p> <p>The IO Streams are now explicitly closed, allowing JSP files to be updated.</p>
CR179636	<p>Due to the limitation of 64K size of Strings, Weblogic Server was failing to compile MultiByte JSP's that comprised of very long Strings. It failed to divide the string for avoiding JVM 64K limitation; therefore, the following Exception, was encountered,</p> <p>java.lang.ClassFormatError: jsp_servlet/_docs/_pas5e/_gp5e0305001 (Illegal constant pool type)</p> <p>WebLogic Server now supports Multi Byte JSPs without any limitation on String Size.</p>

## JTA

Change Request Number	Description
CR125476	<p>When a transaction was in the prepare phase and an XAER_RMERR or an XAER_RMFAIL exception was thrown, the transaction remained pending instead of being rolled back.</p> <p>Now, under these conditions, the transaction is rolled back.</p>
CR188558	<p>When the WebLogic Server was shut down before the timer thread went off, some resources were not being checkpointed to the TLOG. When the WebLogic Server was brought back up, recovery on those resources would not run and any pending transactions would be left pending.</p> <p>Now, when the transaction enlists with a resource, if the resource is coordinated locally and if the resource needs to be checkpointed, it is flagged so that at prepare time, if the transaction is two-phase, the resource gets checkpointed to the TLOG.</p>
CR184941	<p>After a manual JTA migration to back up the server, attempts to boot the crashed WebLogic Server were not successful.</p> <p>Now, after a manual JTA migration to back up the server, if the crashed WebLogic Server is restarted, it boots successfully.</p>
CR184585	<p>Added support for XAResource implementations that possess thread affinity behaviors.</p>
CR134081	<p>When a resource adapter becomes unregistered, either by calling unregister or by a resource failure, all transaction operations intended for that resource adapter fail.</p> <p>Starting with Service Pack 5, if a second resource adapter with the same name is registered, all transactions that were intended for the first resource adapter will now operate on the second resource adapter.</p>
CR176663 CR198213	<p>Transactions that were recovered from the TLOG that had a non-XA participant could not be completed.</p> <p>After a code modification, WebLogic Server now allows these transactions to be successfully completed.</p>
CR130073	<p>When an application issued a number of rollbacks from Tuxedo 7.1 through WTC to WebLogic Server, some Oracle XA connections were leaked and not returned to the pool. XA Debug showed that XA End was not called so the connection was not cleaned up.</p> <p>WebLogic Server now calls XA End, allowing the connections to be returned to the pool and eliminating the out of memory error.</p>

## JTS

---

<b>Change Request Number</b>	<b>Description</b>
CR196085	<p>All public Statement methods are protected by a check that stops a user from doing any more JDBC once a transaction is no longer in progress. In the case of a transaction being rolled back, WebLogic Server itself was calling cancel() on any ongoing statements and the check was blocking WebLogic Server as if WebLogic Server were a user. The following exception was then being thrown:</p> <p>java.sql.SQLException: The transaction is no longer active</p> <p>The check for the cancel() call has been removed. As a result, SQLException is no longer thrown when a transaction is rolled back.</p>
CR189994 CR197412	<p>Upon connection.close, if autocommit was set to false, the updates to the database were getting committed.</p> <p>To avoid this, WebLogic Server now does a rollback first before closing the physical connection during connection pool reset and destroy operations.</p> <p>Now, in order to ensure data integrity, the functionality does not commit by default when closing the physical connection due to a reset / destroy if setAutoCommit(false) is used.</p>

---

## JVM

---

<b>Change Request Number</b>	<b>Description</b>
CR122722	<p>In rare situations on a multi-processor system, a race condition in the Sun JDK would cause a java.lang.Error UnsupportedCharsetException which would terminate the running application.</p> <p>Sun resolved the problem in JDK 1.4.2_05 by redesigning the way they load character sets (see Sun BugParade 4838512). No change was made to WebLogic Server and there are no side effects for WebLogic Server.</p>

---

## Node Manager

<b>Change Request Number</b>	<b>Description</b>
CR184261	<p>Threads were hanging on the Administration Server and the NodeManager when simultaneous start and kill requests were issued for the same server.</p> <p>WebLogic Server now checks for duplicate requests and takes the appropriate action to prevent hung threads.</p>
CR193193 CR202754	<p>NodeManager was not restarting a managed server when all the threads in the default queue were stuck.</p> <p>Now, if all the threads in the default queue are stuck and there are no other application queues, NodeManager detects that the server is in a failed state and restarts the server.</p>
CR189124	<p>For UNIX platforms, the pid file was being created when the NodeManager started a managed server, but the pid was not being written to the pid file.</p> <p>Now, WebLogic Server writes the pid to the pid file for UNIX platforms.</p>
CR184760	<p>The managed server was becoming defunct if it was shut down abruptly through Node Manager on Red Hat Enterprise Linux 3.0-1 because the NodeManager was ignoring the SIGCHLD which was not POSIX-compliant.</p> <p>To avoid this situation, the code has been made POSIX-compliant for Linux OS i686 only because the problem did not occur on any other UNIX flavors.</p> <p>As a result, the managed server processes now exit cleanly and do not become zombie processes in Linux.</p>
CR173733 CR197202	<p>CustomTrustKeyStorePassPhrase is now encrypted so that it no longer appears in clear text in the nodemanager.config file.</p>
CR186592 CR206553	<p>NodeManager was leaking sockets when ...</p> <ol style="list-style-type: none"><li>1. A server start request was sent from the Administration Server to the NodeManager and it was not being properly closed.</li><li>2. Duplicate start/stop requests were sent for the same server.</li></ol> <p>When a socket connection is abruptly broken, WebLogic Server properly closes the socket and its underlying streams.</p>

# Operations, Administration, and Management

<b>Change Request Number</b>	<b>Description</b>
CR127031	<p>When trying to create a JMSQueue or a JMSTopic using the wlconfig ant task, a NullPointerException was seen. The exception appeared only when the parent attribute was used. The following error message was generated:</p> <p><b>BUILD FAILED</b></p> <p>&lt;file:&lt;drive&gt;:/.../build.xml:6&gt;: Error invoking MBean command: java.lang.NullPointerException</p> <p>WebLogic Server now addresses the parent attribute correctly and the NullPointerException no longer occurs.</p>
CR177534	<p>For two reasons, messages were not getting forwarded to the domain log even though managed server log messages were still being written and they should have been forwarded.</p> <ol style="list-style-type: none"><li>1) Using LogManager.readConfiguration and removing and adding WebLogic Server handlers to the root logger, the log was being closed inadvertently.</li><li>2) On the managed server, logging messages with large stack traces that were being forwarded to the domain log were causing the domain log forwarding mechanism to cease forwarding messages.</li></ol> <p>Code has been changed that resolved the problem. Now, messages are not getting forwarded to the domain log as expected.</p>
CR172017	<p>When creating a new server through JMX on a remote server, WebLogic Server no longer throws a NullPointerException if a server with the same name already existed.</p>
CR202252	<p>Unregistering MBeans that had any registered RemoteNotificationListeners was causing a memory leak. As a result, garbage collection was not able to clean up this memory. This eventually led to an out of memory situation.</p> <p>All the listeners on the MBean are now removed before it is unregistered. Now, no memory leak occurs when unregistering MBeans that have any registered RemoteNotificationListeners.</p>
CR197981 CR200953	<p>When deployed to a cluster, the Messaging Bridge was not correctly receiving Notification Listener messages after a server restart.</p> <p>Changes were made so that now the Notifications are correctly delivered and the bridge is properly deployed and discovered.</p>

Change Request Number	Description
CR196774	<p>When a WebLogic configuration property was used while booting WebLogic Server and that property was unknown to the WebLogic Server management subsystem, a message of severity WARNING was being logged.</p> <p>The Message severity was lowered to INFO and the message text was changed to accurately reflect the nature of the message.</p>
CR194477	<p>The WebLogic Server WLCONFIG anttask did not correctly initialize internal data structures to correctly deal with Security MBeans (COMMO MBeans). As a result, creating custom providers via WLCONFIG anttask caused a NullPointerException.</p> <p>A correction to WLCONFIG has been made so that it now correctly processes Security MBeans and creating custom providers via WLCONFIG anttask no longer causes a NullPointerException.</p>
CR194076	<p>When the WebLogic logging level was set to FINE and WebLogic Server was under heavy load, all logging was hanging.</p> <p>Access has been synchronized to the WebLogic Log record cache to improve performance. Now, when the WebLogic logging level is set to FINE and WebLogic Server is under heavy load, logging no longer hangs.</p>
CR182115	<p>The weblogic.Admin tool attempted to retrieve a custom type MBean (commo) if the MBean with the provided name did not exist. This resulted in a misleading audit record.</p> <p>Now, the weblogic.Admin tool throws an <code>InstanceNotFoundException</code> when an MBean with the provided name does not exist instead of attempting to look up or create commo MBeans. As a result, the audit record is no longer misleading.</p>
CR17997, CR248538	<p>Log Rotation was logging messages BEA-170017 and BEA-170018 as <code>Alert</code> type messages when they should be logged as <code>Info</code> type messages.</p> <p>Log Rotation messages BEA-170017 and BEA-170018 were changed so that they are now logged as <code>Info</code> type messages. Any program or behavior that relied on BEA-170017 and BEA-170018 to be an <code>Alert</code> needs to be changed to rely on <code>Info</code>.</p>
CR126191	<p>The DESTROY_POOL command used with the weblogic.Admin tool merely disconnected all the users and suspended the pool. It did not actually delete the pool from the domain configuration as it should.</p> <p>DESTROY_POOL now deletes the pool configuration. Use DISABLE_POOL or SUSPEND to suspend the pool.</p>

Change Request Number	Description
CR191350	<p>The properties:</p> <pre data-bbox="357 458 991 517">-Dweblogic.Stdout=myserver/myserver_stdout.log -Dweblogic.Stderr=myserver/myserver_stderr.log</pre> <p>were generating an unrecognized property warning when used because they were not in the list of recognized WebLogic Server properties. The properties' functionality worked despite the warning message.</p> <p>These properties have been added to the list of known WebLogic Server properties and the warning message no longer appears.</p>
CR191340 CR252299	<p>When piping standard error and standard output to a file by simultaneously using the <code>-Dweblogic.Stdout</code> and <code>-Dweblogic.Stderr</code> options, the standard output was erroneously written into the log file specified using the <code>weblogic.Stderr</code> option, while the standard error was not written into any log files even if the <code>-Dweblogic.Stderr</code> option was used.</p> <p>Now, when piping standard error and standard output files using the <code>-Dweblogic.Stdout</code> and <code>-Dweblogic.Stderr</code> options, both the standard error and the standard output are written to the correct log files.</p>
CR191198	<p>The <code>ErrorDestination</code> parent could not be set to the JMS Queue using the <code>WLConfig</code> and the WL admin scripts.</p> <p>Now, the <code>WLConfig</code> and WL admin scripts work as expected. They are able to set <code>ErrorDestination</code> on a deployed server.</p>
CR175911	<p>The system property, <code>weblogic.jsp.windows.caseSensitive</code>, was generating an unrecognized property warning when used because it was not in the list of recognized WebLogic Server properties. The property's functionality worked despite the warning message.</p> <p>The system property, <code>weblogic.jsp.windows.caseSensitive</code>, has been added to the list of known WebLogic Server properties and the warning message no longer appears.</p>
CR174775	<p>The system property, <code>management.discover</code>, was generating an unrecognized property warning when used because it was not in the list of recognized WebLogic Server properties. The property's functionality worked despite the warning message.</p> <p>The system property, <code>management.discover</code>, has been added to the list of known WebLogic Server properties and the warning message no longer appears.</p>

Change Request Number	Description
CR135573	<p>When a managed server booted in MSI mode, the msi-config.xml file was not being replicated correctly.</p> <p>Now, WebLogic Server checks to see if the server is running in MSI mode each time instead of using a cached value. As a result, the msi-config.xml file is replicated correctly when a managed server boots in MSI mode.</p>
CR134167	<p>EXISTS_POOL did not work as expected. It was looking for runtime mbeans which may or may not exist even if the pool exists. DELETE_POOL, an undocumented command, was the internal implementation to delete a connection pool; however, this feature was documented as DESTROY_POOL externally.</p> <p>The Help menu also displayed some undocumented commands.</p> <p>EXISTS_POOL command implementation now looks for configuration mbeans to check whether the pool exists.</p> <p>DESTROY_POOL command now uses the underlying DELETE_POOL implementation.</p> <p>All the undocumented commands that appeared in the weblogic.Admin help menu are now disabled. Commands include TEST_POOL, REMOVE_POOL, SUSPEND_POOL, SHUTDOWN_POOL, RESUME_POOL, DELETE_POOL. These will not be supported going forward and will not work in 7.0 SP5.</p> <p>The Help menu no longer lists these commands.</p>
CR127563	<p>The system property, net.http.URLStreamHandlerFactory, was not being recognized because it was not in the list of recognized properties.</p> <p>It has been added to the list of recognized properties and is now being recognized.</p>
CR127379	<p>When reconnecting to managed servers, WebLogic Server was using the IP address instead of the hostname that had been specified in the config.xml file. As a result, hostname verification was not working.</p> <p>Now, WebLogic Server uses the hostname to reconnect and hostname verification is working.</p>
CR184080 CR177963	<p>On repeated deploy / undeploy cycles of the EAR with weblogic.Deployer the permanent generation area of the managed server grew until an OutOfMemory error was reported.</p> <p>WebLogic Server now cleans metadata hashmap entries when an Mbean is unregistered. This eliminates the OutOfMemory situation caused by deploying and undeploying repeatedly.</p>

<b>Change Request Number</b>	<b>Description</b>
CR111896	<p>The WL.Admin -batchupdate command did not allow single or double quoted strings in the batch file.</p> <p>The single and double quote characters are now stripped before processing the command so that batch files can have quoted strings.</p>
CR108496	<p>When editing an EJB deployment descriptor from the console, the &lt;global/&gt; tag was being changed to &lt;global&gt;true&lt;/global&gt; in the deployed EJB jar file.</p> <p>An empty tag for getGlobalRole method was added so that validation does not fail after persistence.</p>
CR175474	<p>A code assertion error was thrown when removing a webapp while the server it was deployed to was unavailable.</p> <p>The assertion was removed and processing continues to remove the webapp.</p> <p>WebLogic Server no longer throws an assertion in this particular case.</p>
CR134843	<p>When enforcing the filecount, the code used cached values of the directory contents taken prior to log rotation. These values did not include the latest log file, thus the file count was sometimes one higher than the value set.</p> <p>WebLogic Server now re-reads the contents of the directory after rotation so that the filecount is enforced correctly.</p>
CR186644	<p>The Administration Console was updating the PathURI attribute incorrectly when another attribute, LoadOrder, was modified.</p> <p>PathURI is no longer settable after deployment, therefore LoadOrder modification no longer updates it incorrectly.</p>
CR182686	<p>A ConcurrentModificationException was sometimes thrown while setting System Properties in a startup class or servlet.</p> <p>WebLogic Server has been modified to synchronize the properties and no longer throws the ConcurrentModificationException when setting System properties.</p>
CR177106	<p>WebLogic Server generateConfig generated startup scripts with JAVA_VENDOR SUN instead of Sun.</p> <p>WebLogic Server now generates startup scripts with the correct case for Sun.</p>

Change Request Number	Description
CR176445 CR183762	WebLogic Server threw a NullPointerException when trying to target an application's EJB using nested create commands. WebLogic Server no longer throws a NullPointerException when processing nested create commands.
CR183420	getFullPath was @excluded and was therefore unavailable to clients. Applications can now call applicationMBean.getFullPath() to retrieve the application path on the Administration Server.
CR183836	Caching of log records was causing OutOfMemory conditions.
CR197100 CR187104	WebLogic Server now provides an option so the number of log messages cached is configurable from 0 to infinity. The default is still 500 if this option is not used.
CR183961	The WebLogic Server ant task "nice shutdown" did not have a configurable time delay. A "Delay=" feature was added to WebLogic Server which now allows you to specify the delay in seconds before shutdown.
CR184787	The Administration Console did not display the "replicated_if_clustered" value for PersistentStoreType on the session parameters page of the Deployment Descriptor page. As a result, when the descriptor was persisted, it put in a different value into the descriptor. The Administration Console now displays the value and it is correctly added to the descriptor.
CR185307	The WLConfig ant task used "parent" to set the "Parent" mbean attribute. The WLConfig ant task now uses the correct attribute.
CR126499	WebLogic Server returned an incorrect exit code of 0 when the "ping" command failed. WebLogic Server now returns the correct exit code in the event of an error.
CR121202	User deployment actions sometimes interacted with server-scheduled work to cause a deadlock while logging and saving the Configuration repository. WebLogic Server now has ordered the locks and moved the logging messages outside the synchronization code.
CR112292	weblogic.Admin TEST_POOL was not throwing an exception if the connection failed, and was therefore returning a 0 rather than a 1. WebLogic Server has been changed to make sure that an exception is thrown, in the event that there is a failure in testing the connection.

Change Request Number	Description
CR108002	<p>weblogic.Admin CREATE_POOL used RefreshMinutes even though it was deprecated in 8.1. If you set refreshPeriod through CREATE_POOL, it was not displayed from the Administration Console because the Administration Console retrieved a value from TestFrequencySeconds.</p> <p>WebLogic Server now uses TestFrequencySeconds instead of refreshMinutes when setting attributes on JDBCConnectionPoolMBean via an ant task.</p>
CR101263 CR184396	<p>When running the EXISTS_POOL command on the admin server of a domain with managed servers, the command only returned true if the pool was deployed on the admin server. Since connection pool names must be unique in a domain, this command should have returned true if the pool existed in the domain, not only on the server.</p> <p>WebLogic Server now returns true if a pool with the specified name has been configured in the domain.</p>

## Plug-Ins

---

<b>Change Request Number</b>	<b>Description</b>
CR186148	<p>When the Apache plug-in encountered a missing page, it was returning a 500 error, rather than the correct 503 error.</p> <p>The plug-in now returns the correct error.</p>
CR198530	<p>The If-Modified-Since header was not being sent to WebLogic Server from the Apache plug-in using both non-SSL and SSL.</p> <p>This problem has been resolved.</p>
CR201736	<p>The WL-Proxy-SSL header is no longer missing when using the ISAPI plug-in.</p>
CR199668	<p>Hostname matches were case sensitive, so "HostName" did not match "hostname" when doing a host search.</p> <p>Now, the host search for iisforward.ini proxy is no longer case sensitive. So, Hostname matches are not case sensitive.</p>

---

Change Request Number	Description
CR199446 CR200295	<p>The Apache plug-in did not support regular expressions in the Location and the LocationMatch tags.</p> <p>Now, the Apache plug-in supports POSIX regular expressions.</p> <p>The following Location tag is not valid anymore:</p> <pre data-bbox="357 569 538 595">&lt;Location */app/*&gt;</pre> <p>...</p> <pre data-bbox="357 649 471 675">&lt;/Location&gt;</pre> <p>The tag should be changed to:</p> <pre data-bbox="357 729 518 755">&lt;Location /app/&gt;</pre> <p>...</p> <pre data-bbox="357 808 471 835">&lt;/Location&gt;</pre> <p>Following are some commonly used entries in Location tags:</p> <p>Request matches a certain path:</p> <pre data-bbox="357 930 413 956">/app/*</pre> <p>should be changed to:</p> <pre data-bbox="357 1010 404 1036">/app/</pre> <p>Request starts with a certain path:</p> <pre data-bbox="357 1090 413 1116">/app/*</pre> <p>should be changed to:</p> <pre data-bbox="357 1170 420 1196">^/app/</pre> <p>Request matches a particular mime type:</p> <pre data-bbox="357 1249 413 1275">*/*.jsp</pre> <p>should be changed to:</p> <pre data-bbox="357 1329 397 1355">\.jsp</pre>
CR199045	<p>When multiple object tags were configured for various path values using both SSL and non-SSL configuration, the plug-in was unable to switch correctly between SSL and non-SSL requests.</p> <p>Replaced a global flag (convertDNSSToIP) with a per request flag (moved inside ConfigInfo structure), so that the plug-in now switches correctly between SSL and non-SSL requests.</p>

Change Request Number	Description
CR193447	<p>The cookie did not contain,  , as the delimiter separating primary and secondary information present. As a result, parseJVMID() always returned the primary server information and ignored the secondary server information.</p> <p>The cookie is now tokenized to separate primary and secondary information and then call parseJVMID() for both of the extracted values. Now, parseJVMID() returns both the primary server information and the secondary server information.</p>
CR194464	<p>A call to internal initJVMID() was not updating the state of the server from GOOD to BAD if the connection was refused. As a result, time was wasted trying to reach a server that was already down.</p> <p>Now, if the server is already marked BAD, initJVMID() will skip it and try the next server. Also, initJVMID() updates the state of the server if the connection is refused.</p>
CR192875	<p>The iPlanet Server was crashing if readPostDataIntoFile() threw a new exception from its try catch block.</p> <p>This no longer occurs because readPostDataIntoFile() now returns an exception instead of throwing it if it encounters an error while writing post data to a temp file.</p>
CR191552	<p>The deprecated property, MaxSkips, was replaced by MaxSkipTime. This new property was not being used throughout the code. As a result, the parameter, MaxSkipTime, defaulted to a value of 10 that could not be changed.</p> <p>MaxSkipTime is now used throughout the code and therefore its value can now be changed.</p>
CR189933 CR199045	<p>The WebLogic Server plug-in was not thread safe. The memory address to SrvrInfo array and its size were being passed around and then could be modified by another thread. If they were then modified by another thread, the first thread could end up accessing invalid memory location which could result in seg fault.</p> <p>Now, the WebLogic Server plug-in is thread safe. WebLogic Server makes a read-only copy of SrvrInfo array before passing it around.</p>
CR194141	<p>The Apache plug-in was decoding the URI before passing the request to WebLogic Server even though getRequestURI states that the returned value should not be decoded.</p> <p>Now, the Apache plug-in passes the URI to WebLogic Server without decoding it first.</p>
CR193985	<p>WebLogic Server was logging "creating timer thread in child" messages as [warn] in the log level.</p> <p>Now, these messages are logged as [info] in the log level which better reflects the nature of the message.</p>

<b>Change Request Number</b>	<b>Description</b>
CR100070	<p>It was not easy to maintain the WLogFile for each virtual host in the Apache-WLS plug-in because there was only one global debug flag to enable or disable debugging for all the requests handled by the plugin.</p> <p>Now, it is possible to specify the debugging option at the top most level and overwrite it within each virtual host and/or location tag.</p>
CR190562	<p>Requests were not retried when the plug-in encountered a broken pipe error on Solaris while sending post data to WebLogic Server. WebLogic Server now throws a HALF_OPEN_SOCKET_RETRY exception when sendPostData reports a broken pipe on Solaris</p>
CR189251	<p>Under load, Segmentation errors occurred while retrieving plug-in Properties for a virtual host.</p> <p>Replacing strtok API with strchr, as strtok API is not thread safe, eliminated the errors.</p>
CR182971	<p>ServerList was deleted after every DNSRefreshInterval which resulted in a core dump.</p> <p>WebLogic Server now does a DNS lookup of all the servers in the list and updates the ServerInfo structure if any server has changed from the last time it was checked.</p>
CR095984	<p>When the file size exceeded 30KB, a DNS error message of IE 6.0 was received instead of a 413 message when using the NSAPI plug-in.</p> <p>The behavior of MaxPostSize configuration is now the same with or without a plug-in.</p>
CR180560	<p>The plug-in was not printing the socket information (localhost:localport remotehost:remoteport) to the log file when making a new connection to WebLogic Server.</p> <p>Log messages with the local hostname and local port number are now added when the plug-in makes a new connection to WebLogic Server.</p>
CR186470	<p>When using the IIS plug-in, the creation of a large number of new connections through a firewall resulted in an HTTP status 302, and the connection was closed.</p> <p>WebLogic Server now recycles the connections if the HTTP status code is 302.</p>
CR188811	<p>WLExcludePathOrMimeType did not work correctly when a request had a query string.</p> <p>Requests such as http://webserver:port/weblogic/something.jsp?value=123 were not excluded and requests such as http://webserver:port/weblogic/something.do?name=test.jsp were not forwarded.</p> <p>The plug-in now ignores query strings while checking for excludes.</p>

<b>Change Request Number</b>	<b>Description</b>
CR187578	<p>When KeepAliveEnabled ON was configured in httpd.conf, KeepAliveSecs defaulted to 20. This default setting could not be changed.</p> <p>Code was changed to ensure KeepAliveSecs is configurable.</p>
CR187577	<p>When using multiple Location tags in a VirtualHost tag, the Apache plug-in generated a strange URL if the requests matched two more Locations.</p> <p>The Apache plug-in no longer uses regular expression match, unless specified.</p>
CR187282	<p>Because the plugin did not follow a part of the HTTP1.1 specification, which states that if a request/response contains both a Content-Length header as well as a Transfer-Encoding: Chunked header, the Content-Length header MUST be ignored, there was a unique scenario involving a recycled connection from the pool that sometimes caused an error.</p> <p>WebLogic Server now returns connectLength as -1 if CTE is on.</p>
CR186148	<p>When the Apache plug-in encountered a missing page, it was returning a 500 error, rather than the correct 503 error.</p> <p>The plug-in now returns the correct error.</p>
CR185668	<p>When using the Apache plug-in to proxy to multiple clusters using MatchExpressions, the PathTrim attribute was failing to trim off the segment of the url used to direct the request.</p> <p>Reimplementing MatchExpressions parsing without using the strtok API corrected the problem.</p>
CR185089	<p>The IIS plug-in was sending an Http status code of 500 (internal Server Error) when it encountered a WRITE_ERROR_TO_CLIENT exception due to a connection closed by the client.</p> <p>The IIS plug-in now sends Http status code of 503 when a WRITE_ERROR_TO_CLIENT exception is caught.</p>
CR183390	<p>WebLogic Server was throwing an exception from inside the catch block which sometimes caused iPlanet to fail.</p> <p>WebLogic Server no longer throws an exception from the catch block.</p>
CR183311	<p>When Apache was stopped while using a single-thread multi-process module, it would try to stop the timer thread first. This timer thread never existed, thus a core dump occurred.</p> <p>WebLogic Server no longer creates timer threads when Apache is being used with a single-thread multi-process module.</p>

<b>Change Request Number</b>	<b>Description</b>
CR183188	<p>The ISAPI plug-in was unable to handle requests with the Transfer-Encoding header set to chunked.</p> <p>Functionality was added to enable ISAPI to handle such requests.</p>
CR182434	<p>Headings passed to rq-&gt;srvhdrs were not entirely in lower-case instead of mixed case.</p> <p>Content-type, content-length and transfer-encoding headers are now passed to NSAPI entirely in lower case.</p>
CR180724	<p>The initial cookie was created through web server one and sent to cluster one. When it hit the application again it went through web server two and instead of being directed to cluster 1 it went to cluster 2 and created a new session.</p> <p>The WLCrossOverEnabled functionality now works correctly in WebLogic Server.</p>
CR180417	<p>If a cookie was part of the POST data then the plug-in would corrupt the post data while extracting the cookie.</p> <p>Code was added to fix the cookie extraction from Post Data.</p>
CR136968	<p>Weblogic Server was not accepting more than one header when the response.addHeader method was used.</p> <p>The plug-in now allows WWW-Authenticate to have multiple values.</p>
CR178792	<p>(Apache) HTTP requests can contain either one of the following headers: Content-Length or Transfer-Encoding</p> <p>Requests with a Transfer-Encoding header set to "chunked" were failing with an IO error.</p> <p>Code was added to support requests using the Transfer-Encoding header set to "chunked".</p>

## RMI/RMI-IIOP

Change Request Number	Description
CR176676	<p>RMI timeout enables an RMI client making a remote call to return even when the remote method that it invoked has not returned from the called server. In other words, it does not wait to be notified on getting a response. Sometimes, due to the nature of the application, the response might take a long time to arrive. In such cases, the waiting thread on the caller is blocked. RMI timeout will provide a handle (during <code>weblogic.ejb/weblogic.rmic</code>) in order that a timeout value per remote method can be specified.</p>
CR199420	<p>Using <code>AsyncMessageSender</code> was resulting in stuck threads when sending large messages. Under sustained load, a thread was stuck indefinitely.</p> <p>Thread sending has now been batch pined. As a result, threads are no longer stuck when sending large messages using <code>AsyncMessageSender</code>.</p>
CR189042	<p>Now, WebLogic Server can proxy IIOPS requests using the thin client.</p>
CR192252	<p>For hardware load-balancers in conjunction with IIOP, the system property, <code>reconnectOnBootstrap</code>, is now working correctly in the BEA ORB.</p>
CR192648	<p>The thin client usage of the <code>SocketImpl</code> class was causing network-related failures due to a problem in the way J2SE determines the implementation of the <code>SocketImpl</code> class.</p> <p>The thin client has been changed to work around the J2SE problem. As a result, clients can now correctly connect through a firewall.</p>
CR195105	<p>WebLogic Server was sending the wrong port information so that while trying to download the stubs, the client was trying to connect to the wrong port. As a result, thin client could not be run using <code>wlclient.jar</code> with <code>t3s</code>.</p> <p>Changes were made in the <code>CodebaseComponent</code> class so that the server now sends the correct address and port information to the client. As a result, thin client can work through a firewall.</p>
CR194465	<p>During failover, when using <code>thinclient</code> and tunneling, the applet was trying to resolve the public address of the backend servers which resulted in a security exception.</p> <p>Now, WebLogic Server does not resolve public addresses during tunneling. As a result, there is no security exception during failover when using <code>thinclient</code> and tunneling.</p>
CR190380	<p>IIOP connections could appear to randomly fail after initial use. These connections are now cleaned up correctly and therefore no longer appear to randomly fail.</p>

<b>Change Request Number</b>	<b>Description</b>
CR184662	Serializing the stateful session bean (SFSB) handle no longer results in a NO_IMPLEMENT exception when using the IIOP protocol.
CR188229 CR196372	JDBCWrapper objects were lying in the heap even after the connection, statement, and result set objects were closed. The JDBCWrapper objects are now GC'ed after the connection, statement and result set objects are closed.
CR177353	<p>When a application client code cached the remote stub and invoked a remote method on a SLSB deployed to a cluster, the behavior was that each call refreshed the list which tracked the cluster nodes where the remote object is available. This list was used to failover the calls if any of the node failed with a recoverable exception. The issue here was that failover did not work when the entire cluster was restarted while the application client had cached the stub from a previous invocation.</p> <p>The retry logic in the failover algorithm was incorrect. This logic originally allowed n-1 number of retries in a cluster with n nodes. When the entire cluster restarted, the cached stub would have stale list. And the retry logic scanned though the stale list and exhausted all the retry attempts. In the last attempt it would have potentially refreshed the list. Even though the client side stub now had a new copy of the list it did not attempt to failover as it has already reached n-1 attempts limit.</p> <p>The remote stub cached in the application client now ensures it refreshes the list only when remote method invocation fails on all the nodes in the existing list. Then stub is given one last chance for failover if the list got refreshed. If this last chance does not succeed the stub will throw an exception to the application otherwise failover will continue to work as advertised transparently.</p>
CR189042	<p>IIOPS requests were unable to be proxied through an outbound webfilter such as Squid. These webfilters only allowed HTTP and HTTPS requests through.</p> <p>WebLogic Server now allows the thin-client to create a proxy socket through the webfilter using the CONNECT command. IIOPS requests can now use this proxy socket.</p>
CR179595	<p>During redeploy, the old finder was reused. Hence the new skeleton, according to new interfaces, was not generated. Instead, the old skeleton was used for new remote interfaces.</p> <p>WebLogic Server now creates a new finder when the classloader is bounced.</p>

Change Request Number	Description
CR183490 CR192555	<p>When a stub was deserialized the server did not call connect on the stub which resulted in exception stating "Delegate not set".</p> <p>In addition a problem occurred when a stub was obtained by connecting to a cluster. After deserialization the information that the stub was connected to the cluster was lost. The deserialized stub only remembered the last server to which it was connected.</p> <p>WebLogic Server now extends a class from <code>com.sun.corba.se.internal.javax.rmi.CORBA.StubDelegateImpl</code>. In the new class derived from <code>StubDelegateImpl</code> WebLogic Server overrides <code>writeObject</code> and <code>readObject</code>. while writing, the object WebLogic Server also writes the <code>clusterURL</code> and while reading the object, WebLogic Server reads the <code>cluster url</code> and call <code>connect</code> on the deserialized stub.</p> <p>A system property must be set on the client side in order to access this new functionality. The property is named <code>javax.rmi.CORBA.StubClass</code>. so the client side script might include something like this:</p> <pre>-Djavax.rmi.CORBA.StubClass=weblogic.corba.client.iiop.WLStubDelegate</pre>
CR131692	<p>When throwing a derived exception from the <code>ejb</code> method, the <code>iiop</code> layer was adding the Full Value Description for the derived exception but not for the base exception. When the client came back to get information about the base exception the server did not have it and threw an exception.</p> <p>Code has been added to ensure that when creating a Full Value Description for the derived class, the Full Value description of the base classes is created as well.</p>
CR189067	<p>An applet hosting an RMI object was getting DGC'd after four minutes which is the distributed garbage collection (DGC) timeout period. The applet was unable to download the relevant <code>rtd</code> file for <code>DGCServer</code>, thereby allocating it an <code>oid</code> of <code>&gt;256</code>. As a result, the leases were not being renewed for the RMI object and the applet was getting DGC'd.</p> <p>To fix this, WebLogic Server now allows the <code>ClasspathServlet</code> to service <code>.dtd</code> and <code>rtd</code> files.</p>

# Security

<b>Change Request Number</b>	<b>Description</b>
CR180841	<p>When the <code>areWebAppFilesCaseInsensitive()</code> method was added to the Security Service Manager, the calculation of the case-setting changed for client processes. As a result, the configuration and listing of security roles and security policies for a URL resource was failing. This problem has been resolved.</p>
CR185283 CR197577	<p>When using the T3 or IIOP protocol, WebLogic Server was not propagating the Security context (<code>javax.security.auth.Subject</code>) between WebLogic Server instances.</p> <p>The WebLogic Credential Mapping provider is now called for each outbound CSIV2 session that is established from a server. The Credential Mapping provider uses the inbound CSIV2 principal instead of the <code>AuthenticatedSubject</code> principal to propagate the security context. Note the WebLogic Credential Mapping provider does not have access to the credentials for the inbound principal.</p> <p>If the credentials are required, a custom Authentication provider needs to be written to add them to the Subject so that the WebLogic Credential Mapping provider can access them.</p>
CR185792	<p>When an LDAP connection was dropped by the LDAP server, WebLogic obtained a new connection. However, the stale connection was not getting cleared.</p> <p>The stale LDAP references are now cleared correctly when using the Compatibility realm.</p>
CR193305	<p>When querying the <code>JDBCConnection</code> MBean from a <code>ServletContextListener</code> and then deploying the associated EAR from the Administration Console, a <code>NoAccessRuntime</code> exception was thrown even though the application was successfully deployed.</p> <p>This problem has been resolved.</p>
CR177522	<p>When a context was opened during a <code>.jms</code> call, the subject was pushed onto a stack of subjects. If the context was not closed, the subject was not removed, leaving an extra subject on the stack.</p> <p>When the application continued to process and at some point accessed a protected resource, the extra subject was taken from the stack as the current subject. In some cases, it did not have the required privileges, so access to the resource was denied. Closing the context removed the subject from the stack, restoring the correct subject to the top of the stack.</p>

Change Request Number	Description
CR186186	<p>The Certicom code was handling UTF8 incorrectly. It was treating the bytes of a String as individual characters causing a NullPointerException when Scandinavian characters are used.</p> <p>WebLogic Server now processes UTF8 correctly.</p>
CR182860	<p>When Web browsers sent plain text through the secure port, SSL misunderstood the message header and waited for more data before finishing the message. This delay caused problems with the SSL handshake. The server was using 100% of its CPU usage trying to complete the SSL handshake. This problem was caused by an arithmetic error on the number of bytes read. This arithmetic problem has been solved. Clients that send plaintext messages to the secure port now receive errors earlier and the server will not hang because of 100% CPU utilization.</p>
CR186916	<p>Two aspects of SubjectUtils.isUserInGroup were problematic.</p> <ul style="list-style-type: none"> <li>• Getting only Principals of a specific class required evaluating each Principal while building a set of Principals, then iterating through them to find if any Principal was in the group being searched for. This problem effectively caused two iterations through the Principals when one could be used to get all Principals of the Subject and then iterating for the group name.</li> <li>• Getting an AuthenticatedSubject from the javax.security.auth.Subject interface and searching for the Subject in a group could not be done unless the Subject had been authenticated. This problem meant additional cycles to authenticate the Subject and determine to which groups the Subject belonged.</li> </ul> <p>RESOLUTION:</p> <p>All Principals are now gotten from the Subject and are iterated through once looking for the specified WLSGroup.</p> <p>A second method has been added with an AuthenticatedSubject as the first parameter. Both public isUserInGroup () methods were changed to call an internal isUserInGroup () method that works with a set of Principals rather than any Subject. This method eliminates the need for conversion from a Subject to an AuthenticatedSubject.</p>
CR185315 CR199865	<p>The implementation of LDAP server in WebLogic Server had connection handlers each of which ran in its own threads. However, the implementation was not starting and stopping the threads properly therefore thread memory was not being recovered.</p> <p>The implementation was changed so that the connection handlers no longer run as threads. This change allows normal memory recovery to take place.</p>

Change Request Number	Description
CR172912	<p>If two trusted CA certificates had the same Subject name, WebLogic SSL implementation was only using the first trusted CA certificate. This often resulted in an SSL client failing the peer certificate validation check.</p> <p>This problem has been resolved. Now if there are two trusted CA certificates with the same Subject name, the SSL implementation finds the appropriate trusted CA certificate for the peer certificate presented.</p>
CR132634	<p>Memory Leaks were detected when using DirContext.close with the embedded LDAP server. This problem has been resolved.</p>
CR179633	<p>When a client program was trying to establish a mutually-authenticated SSL connection to WebLogic Server, it was able to establish the connection and perform its task only once. Subsequent attempts to establish a connection were failing.</p> <p>This problem has been resolved.</p>
CR182702	<p>When the Web Application was restricted with (*) in the web.xml file there was no constraint on access to the root directory (/). For example, if the application was foo and policy was defined for (*) in the web.xml file, the result was that /foo was not protected.</p> <p>Now, WebLogic Server constrains access to the root directory (/) when the webapp is restricted with (*) in the web.xml file.</p>
CR201690	<p>The getLatestFailure() method no longer throws a NoSuchElementException.</p>
CR124562	<p>WebLogic log was ignoring the Principals that extend WLSUser. Therefore, sometimes the wrong Principal was being displayed in the user field.</p> <p>Now, WebLogic log displays the custom Principals that extend WLSUser.</p>
CR125177	<p>WebLogic Server was not checking for group circularity in weblogic.security.providers.authentication.LDAPAtnLoginModuleImpl. As a result, so many LDAPConnections were being created when the backend LDAP group had itself as a member that the server would crash.</p> <p>Now, WebLogic Server has a flag called IgnoreDuplicateMembership. When checked, this flag warns if a group is being added that has already been added to the list of groups a user belongs to. In this case, WebLogic Server will then ignore the duplicate group. By default, the flag is unchecked.</p> <p>Although this flag works in production mode, it is recommended that it only be used in development mode due to its affect on performance in production mode.</p>

<b>Change Request Number</b>	<b>Description</b>
CR121761	<p>The ParsePolicies class was not parsing some special permissions properly.</p> <p>This problem has been resolved.</p>
CR191229	<p>When Certificate Authentication was configured, Authorization was occurring even for unprotected resources. As a result, access for anonymous users was being denied.</p> <p>Now, when Certificate Authentication is configured, Authorization no longer occurs for unprotected resources.</p>
CR196578	<p>When WebLogic Server 6.1 and WebLogic Server 8.1 interoperated, an Authenticated User sent to a 8.1 server instance was not honored properly. As a result a java.rmi.AccessException was thrown.</p> <p>The Authenticated User was changed to an Authenticated Subject.</p> <p>Now, WebLogic Server 6.1 can interoperate with WebLogic Server 8.1 without a java.rmi.AccessException being thrown.</p>
CR196410	<p>WebLogic Server was not honoring an authenticated Subject when a WebLogic Server security realm interoperated with a Compatibility security realm.</p> <p>By default, WebLogic Server does not honor authenticated Subjects coming over the wire. The WLS User Principal Allowed attribute has been added to the WebLogic Server Administration Console. When the attribute is enabled in the WebLogic Authentication provider, WebLogic Server honors authenticated Subjects coming in over the wire.</p>
CR191419	<p>The weblogic.security.X500Name class now implements the Serializable method.</p>
CR190679	<p>The persisted authenticated Subject was reusable in a second session because the validation process did not detect the configuration change.</p> <p>This problem was resolved by providing new functionality that re-authenticates an authenticated Subject presented during a request after a specified length of time.</p>
CR184203	<p>When a server joined a cluster, it obtained information from other servers in the cluster for purposes of synchronization. If the cluster was configured to use only SSL ports and required client certificates to establish the SSL connection, the synchronization was failing. The problem occurred because the new server (the SSL client) was not presenting a certificate to the other servers in the cluster.</p> <p>This problem has been resolved.</p>

<b>Change Request Number</b>	<b>Description</b>
CR180729	<p>When using the embedded LDAP server, membership information from both the LDAP etc/group and etc/passwd directories was not processed when group membership was determined.</p> <p>WebLogic Server now combines and retrieves the membership information specified in the etc/group and etc/passwd directories.</p>
CR172261	<p>The Certicom implementation of the SSL protocol in WebLogic Server uses a class that is synchronized by the JDK. This synchronization caused the SSL session threads to hang.</p> <p>The SSL implementation now uses a class that is synchronized outside of the JDK. This change resolved the SSL session thread problem.</p>
CR199331	<p>The <code>weblogic.servlet.security.CertificateServlet</code> interface is deprecated in this release. Therefore, the Certificate Request Generator is also deprecated. Use the <code>keytool</code> utility from Sun Microsystems in place of the Certificate Request Generator servlet.</p>
CR112486 CR199852	<p>The <code>CredentialMapper</code> had insufficient privileges to modify the user credential mappings in all cases.</p> <p>WebLogic Server now has sufficient privileges to modify the user credential mappings in all cases.</p>
CR177523 CR189057	<p>When proxying a connection, the proxy hostname was being validated during hostname verification rather than the target system hostname, causing a validation error.</p> <p>WebLogic Server now validates the target system hostname.</p>
CR132656	<p>When the flag <code>-Dweblogic.security.URLResourceCaseMapping=off</code> was used during startup, a warning message was displayed even though the server started correctly.</p> <p>WebLogic Server no longer displays the incorrect warning.</p>
CR176306	<p>The muxable socket for the embedded LDAP server was not correctly calculating the message length. The result was the data in the embedded LDAP was not replicated in a cluster of managed servers.</p> <p>WebLogic Server now calculates the available bytes and the LDAP messages correctly.</p>
CR180972	<p>Performance of outbound SSL connections was slowed compared to JSSE because WebLogic Server created a new <code>SSLConnectionFactory</code> for each outbound request.</p> <p>WebLogic Server now caches the <code>SSLConnectionFactory</code> when there is no certificate for the request.</p>

<b>Change Request Number</b>	<b>Description</b>
CR181263	<p>WebLogic Server was ignoring the <code>weblogic.management.username</code> command-line property. The only way to get the username was to look in the <code>boot.properties</code> file or to prompt the user. Code was added so that WebLogic Server can obtain the username by using the <code>weblogic.management.username</code> command-line property when no <code>boot.propertiesfile</code> is available.</p> <p>WebLogic Server uses the following process to obtain the username:</p> <ol style="list-style-type: none"> <li>1. If it exists, look in the <code>boot.properties</code> file.</li> <li>2. Obtain the username by using the <code>weblogic.management.username</code> command-line property.</li> <li>3. Prompting the user for the username.</li> </ol>
CR182006	<p>Certificate verification was failing because the <code>SubjectAlternativeName</code> extension to the <code>x509</code> certificate was marked as critical.</p> <p>WebLogic Server now allows <code>x509</code> certificates with this extension marked as critical to be verified during the SSL handshake.</p>
CR185438	<p>Connections to external LDAP servers were sometimes dropped if a load balancer or firewall was configured between WebLogic Server and the external LDAP Server.</p> <p>WebLogic Server now automatically refreshes the connections if they are dropped and user requests complete if the connection is dropped during the request.</p>
CR135587	<p>WebLogic Server only supported the use of the <code>weblogic.system.BootIdentity</code> file with the <code>SHUTDOWN</code> and <code>FORCESHUTDOWN</code> operations of the <code>weblogic.admin</code> utility.</p> <p>The <code>weblogic.admin</code> utility has been enhanced to support the use of the <code>weblogic.system.BootIdentity</code> file with all its operations.</p>
CR175966	<p>Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04-72.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04-72.00.jsp</a>.</p>

# Servlets

<b>Change Request Number</b>	<b>Description</b>
CR188348	<p>The HttpURLConnection in WebLogic Server always sent the Connection: Keep-Alive header(or Proxy-Connection: Keep-Alive header) even in cases where http.keepAlive was disabled.</p> <p>Keep-Alive can now be disabled with the -Dhttp.keepAlive=false property.</p>
CR134351	<p>The flag, weblogic.http.client.defaultReadTimeout, has been introduced to set the default readtimeout on WebLogic Server's HttpURLConnection.</p> <p>When this flag is set, every new HttpURLConnection created on the JVM will have this value set as default. The value can be overridden by setTimeout or setReadTimeout calls on the specific connection. When this flag is not used, the default value is -1. The value should be provided in milliseconds.</p>
CR190652	<p>WebLogic Server no longer throws an exception when making a HEAD request that invokes RequestDispatcher.</p>
CR195698	<p>Replacing the old WAR modules with new ones in the external_stage area was causing a 404 error.</p> <p>This problem was resolved by staging the actual application in the temp folder.</p>
CR194121	<p>When ChunkTransferDisabled was set to true in ChunkUtils the java.lang.NullPointerException was being thrown.</p> <p>This problem has been resolved.</p>
CR179800 CR181640 CR173782	<p>Some examples in the final version of the JavaServer Faces v1.0 Reference Implementation fail to preload on startup with the following ServletException:</p> <pre>&lt;Mar 16, 2004 10:31:09 AM MST&gt; &lt;Error&gt; &lt;HTTP&gt; &lt;BEA-101216&gt; &lt;Servlet: "Faces Servlet" failed to preload on startup in Web application: "jsf-guessNumber". javax.servlet.ServletException</pre> <p>The problem was fixed by adding logic during the Web application startup for registering servlet listeners, which could be defined in .tld files in the lib folder of the application.</p>

Change Request Number	Description
CR175097 CR173748	<p>During the process when Webapp deployment extracts the web-inf/lib JAR files to the .wlnotdelete directory, the time stamps of the JAR files were not being preserved. As a result, JSP compilation was failing.</p> <p>Now, WebLogic Server preserves the timestamp of the JAR file when it is extracted to the .wlnotdelete directory.</p>
CR199958	<p>If the context-root was not present in the application.xml file (for an EAR) or the weblogic.xml file (for a given Web Application), WebLogic Server derived the context-root from the name of the WAR file (minus .war) or the name of the root directory if the Web Application was exploded. If the derived context-root was already in use by an existing Web Application, WebLogic Server appended a counter (i.e., /foo-1, /foo-2) in order to make it unique.</p> <p>For example, if /root/apps/app1/foo.war was deployed, and if the application.xml or the weblogic.xml file had not specified the context-root, the context-root was derived as /foo. Then, if another application from /root/apps/app2/foo.war was deployed, the context-root was derived as /foo-1 to avoid context-root conflict.</p> <p>This mechanism was convenient because it allowed deployment of a WAR file with the same name multiple times. However, it caused problems during cluster deployments.</p> <p>WebLogic Server 8.1 SP4 and later releases no longer have this counter mechanism. Now, if context-root clash occurs, WebLogic Server does not allow the application to be deployed. So, it is now necessary to specify a unique context-root in the application.xml file or the weblogic.xml file.</p>
CR106348	<p>A failure in the listener left the Web Application in an unstable (deployed) state instead of undeploying the Web Application.</p> <p>When the listener fails (deployment is not completely successful), the Web Application deployment is rolled back.</p>
CR197121	<p>Now WebAppServletContext.getResourcePaths() returns both directories and files under the web-app root directory as well as any virtual directories configured in the weblogic.xml for the web-app.</p>
CR172672	<p>When HttpClusterServlet tried to reuse a recycled connection that WebLogic Server had already closed, it would get a SocketException and mark the WebLogic Server as BAD.</p> <p>Now, if HttpClusterServlet gets a SocketException when using a recycled connection, it will make another attempt to connect to the same WebLogic Server with a new connection.</p>

Change Request Number	Description
CR175651	<p>To enable the weblogic server to finish inflight requests for a WebApp during undeployment or shutdown of the server, use the following element in weblogic.xml:</p> <pre data-bbox="357 491 1016 591">&lt;container-descriptor&gt;   &lt;inflight-request-timeout-secs&gt;120&lt;/inflight-request-timeout-secs&gt; &lt;/container-descriptor&gt;</pre> <p>The property, <code>inflight-request-timeout-secs</code>, indicates the number of seconds the server will wait in seconds. By default, the server will not wait for inflight work.</p>
CR200043	<p>Locale variants passed by the browser were causing unexpected results in validation rules in Weblogic Server. The locale header parsing logic was not considering the variant part of it. Now, WebLogic Server considers the locale variant while parsing the language header. The language header can be in the following format:</p> <pre data-bbox="357 831 637 855">lang-country-variant;q=weight</pre>
CR199958	<p>The context root display for Web Application Modules sometimes became corrupted as the number of Web Application Modules grew.</p> <p>Now, WebLogic Server no longer supports deploying multiple applications with the same context root. Instead, there is an index appended to the context root as in the following example:</p> <p>Instead of the context root: <code>/test</code></p> <p>Now, the context roots are: <code>/test-1</code>, <code>/test-2</code>, etc.</p> <p>Now, the context root display for Web Application Modules does not become corrupted as the number of Web Application Modules grows.</p>
CR198536	<p>If the WebLogic Server failed unexpectedly, the <code>access.log</code> file sometimes lost the ELF header because the ELF header was not being flushed immediately when a log had been rotated.</p> <p>Now, WebLogic Server flushes the ELF headers immediately after rotating the log. As a result, the <code>access.log</code> file no longer loses the ELF header.</p>
CR185289	<p>An infinite loop sometimes occurred in <code>ChunkOutput.writeStream</code> if the auto flush was <code>'false'</code> and the content length was greater than the buffer size limit.</p> <p>This problem was resolved by adding code to reset the auto flush flag for each new request.</p>

Change Request Number	Description
CR191946	<p>The WebLogic Server WebApp Container was not initializing security for Weblogic Internal WebApps. As a result, the proxy header, WL-Proxy-Client-Cert, was getting ignored.</p> <p>Now, the WebLogic Server WebApp Container initializes security for Weblogic Internal WebApps. The proxy header, WL-Proxy-Client-Cert, no longer gets ignored.</p>
CR191427	<p>WebLogic Server was logging the query string twice if the cs-uri field was specified.</p> <p>Now, WebLogic Server does not log the query string twice if the cs-uri field is specified.</p>
CR184997	<p>This CR is regarding HTTP Log Rotation configured by date option.</p> <p>The HTTP Log Rotation was not rotating the log files at rotation periods if there was no activity in that period. Logic was added so that rotation always occurs at the rotation period whether activity is present during that period or not.</p>
CR191347 CR178729	<p>When a Web App was deployed on a managed server and the managed server was down, if the Web App was deleted and then re-configured using the Console, it was not getting re-deployed to the managed server.</p> <p>Now, the new version of the Web App is re-deployed on the managed server when it has been re-configured while the managed server was down.</p>
CR177095	<p>When using weblogic.appc and if the application code was using the context classloader, WebLogic Server was throwing a ClassNotFoundException for classes in the WEB-INF\lib directory.</p> <p>WebLogic Server no longer throws a ClassNotFoundException under these circumstances.</p>
CR184326 CR188237	<p>HttpClusterServlet was writing to the internal stream and ignored the case when client had a custom response wrapper, which used a custom outputstream.</p> <p>WebLogic Server now correctly uses the custom output stream rather than the response outputstream when the client has a custom response wrapper.</p>
CR185885	<p>During graceful shutdown of the server, the HTTP container was returning 404 when the webapp was unregistered, rather than returning 503 Service Unavailable. This prevented the plug-in from attempting to failover.</p> <p>WebLogic Server now returns a 503 status code when the server is suspended.</p>
CR181754	<p>Even when the WebLogic Server was in suspended state, if a new request was sent to a secured resource, a new session was created for requests with secure resources.</p> <p>WebLogic Server now checks for a suspended state before redirecting the request.</p>

<b>Change Request Number</b>	<b>Description</b>
CR181747 CR189395 CR200944	<p>Weblogic Server was not accepting more than one value for the header WWW-Authenticate header when the response.addHeader method was used.</p> <p>The Weblogic Server now allows WWW-Authenticate to have multiple values.</p>
CR179941	<p>When OptimisticSerialization is turned on, WebLogic server does not serialize-deserialize context and request attributes upon getAttribute(name), when a request is dispatched across servlet contexts.</p> <p>To avoid ClassCastExceptions, you must make sure that the attributes common to the webapps are scoped to a common parent classloader; such as Application scoped or placed in the system classpath if the webapps do not belong to the same application.</p> <p>When OptimisticSerialization is turned off (default value) WebLogic server serializes-deserializes context and request attributes upon getAttribute(name) to avoid ClassCastExceptions.</p> <p>To turn on Optimistic Serialization use the command line flag: "-Dweblogic.servlet.optimisticSerialization=true"</p>
CR177730	<p>wl_authcookie was not being forwarded for single signon.</p> <p>WebLogic Server now reuses wlauthcookie until the session is explicitly logged out or invalidated. By reusing the wlauthcookie across webapps, the single sign on remains valid even if a new user logs in to the first webapp.</p>
CR175781	<p>If HttpURLConnection.getOutputStream() for weblogic.net.http.HttpURLConnection was called more than one time the previous content was deleted from the OutputStream. Data previously written to that OutputStream was lost.</p> <p>WebLogic Server now checks to see if there is a connection, and does not reset the stream if a connection exists.</p>
CR174501	<p>When multiple Web Applications participated in a single signon, the sessions in some of the Web Applications were not invalidated.</p> <p>WebLogic Server now correctly invalidates the sessions in all Web Applications.</p>
CR172088	<p>WebLogic Server returned the HTTP 500 error for requests when the request contained a corrupt If-Modified-Since Http Header.</p> <p>WebLogic Server now ignores the If-Modified-Since Header, if it is malformed.</p>

Change Request Number	Description
CR172077 CR193354	<p>Deployment of installed applications did not obey module order in application.xml while deploying the webapps.</p> <p>During server start, ServletInitService.resume called the preloadResources on the deployed webapps but the Servlets were not necessarily invoked in the order the webapps had been deployed.</p> <p>During server start, preload Servlets are now invoked in the same order as the webapps are deployed.</p>
CR136410	<p>The HTTP access log rotation by size was working incorrectly on a multi-processor machine. After a number of log files were created, the rotation stopped working correctly, and eventually the rotation began overwriting the same two generations of the file.</p> <p>The comparison function was modified, and WebLogic Server now creates new log files as needed.</p>
CR184745	<p>When DirectoryIndexEnabled was used and there were multi byte named files in the target directory, The directory index page became corrupted and was not retrieved correctly.</p> <p>WebLogic Server now has a java-iana map for GBK and MS950, and retrieves multi byte named files correctly.</p>
CR185289	<p>An infinite loop sometimes occurred in ChunkOutput.writeStream if the auto flush was 'false' and the content length was greater than the buffer size limit.</p> <p>This problem was resolved by adding code to reset the auto flush flag for each new request.</p>
CR108736 CR191543	<p>When a user threw an exception wrapped in a ServletException while making a call to init, The exception cause for the ServletException was lost and the line number printed for the ServletException was incorrect.</p> <p>WebLogic Server now handles this correctly. As a result, the Exception log prints the correct line number and the root cause of the ServletException.</p>
CR087857	<p>When the character encoding was set by JSP pages or servlets, the ServletOutputStream.write(int) method, which takes int type as its argument, received the data encoded using the specified charset encoder.</p> <p>WebLogic Server no longer encodes the binary data when ServletOutputStream.write(int) is called.</p>

## SNMP

---

<b>Change Request Number</b>	<b>Description</b>
CR194161	<p>Whenever the SNMP service was shut down, a serverDown trap was sent for each of the configured servers in the domain even if the managed servers were running.</p> <p>Now, when the SNMP service is shut down and the managed servers are running, a serverDown trap is sent only for the Admin server and only if the SendAutomaticTrapsEnabled attribute is true.</p>
CR083630	<p>Certain strings in SNMP traps and notifications were not localized. If the string contained multi-byte characters, it was not being displayed correctly.</p> <p>String values have been localized and internationalized to support multi-byte characters. Now, WebLogic Server displays Japanese characters correctly in the log message trap.</p>
CR190218	<p>Autocast was failing for double type attribute values that had a return of type Integer.</p> <p>WebLogic Server now correctly does the conversion between the data types before returning the value. As a result, Autocast no longer fails for double type attribute values that have a return of type Integer.</p>

---

## Tools

---

<b>Change Request Number</b>	<b>Description</b>
CR183190	<p>There is no longer a java.sql.SQLException on package's dynamicSections.</p>
CR136334	<p>When using the BEA JDBC Driver under load, the following error was occurring: java.lang.OutOfMemoryError: unable to create a new native thread</p> <p>This problem has been resolved.</p>
CR124030	<p>Opening an application.ear file in Marathon, but not changing anything, resulted in an invalid descriptor being written out, and thus the application no longer passed appc.</p> <p>Applications can now be opened in Marathon without being corrupted.</p>

---

## Web Services

<b>Change Request Number</b>	<b>Description</b>
CR199783	<p>WebLogic Server attempted a retry when a Webservice client set <code>weblogic.webservice.rpc.timeoutsecs</code>.</p> <p>Now, WebLogic Server no longer attempts a retry when a Webservice client sets <code>weblogic.webservice.rpc.timeoutsecs</code>.</p>
CR199190	<p>When an element that had over 255 child elements was defined in the schema of Web Service Description Language (WSDL), WebLogic Server was failing to compile the bean source that clientgen had generated.</p> <p>Now, WebLogic Server no longer generates a constructor that takes more than 255 parameters, thus eliminating the problem.</p>
CR198996	<p>WebLogic Server was throwing a <code>SOAPFaultException</code> for some non-WebLogic WebServices when a Content-type header in a SOAP message had single quotes around the charset as in the following example:</p> <p>Content-Type: text/xml; charset='utf-8'</p> <p>Because the single quotes around charset are not necessary, the single quotes have been removed and the <code>SOAPFaultException</code> that resulted from the use of the single quotes in charset is no longer thrown.</p>
CR197698	<p>As a result of a client-side code change, Webservice HTTPS client no longer fails with a <code>java.net.UnknownHostException: null</code>.</p>
CR197241	<p>WebLogic Server was unable to sign the <code>webserviceclient+ssl.jar</code> file using jarsigner because a duplicate class was present in the file. The following exception was being thrown:</p> <p>jarsigner: unable to sign jar: java.util.zip.ZipException: duplicate entry: weblogic/utills/StringUtils.class</p> <p>The duplicate entry in the jar file was removed. As a result, WebLogic Server can now sign the <code>webserviceclient+ssl.jar</code> file using jarsigner with no exception being thrown.</p>
CR195817	<p>Setting the web service client's timeout using the <code>weblogic.webservice.binding.BindingInfo.setTimeout()</code> method was not having any effect on the client when HTTPS was used.</p> <p>Now, this method does have an effect on the client when HTTPS is used.</p>

Change Request Number	Description
CR195535	<p>The URI in the web-services.xml file generated from wsdl2service was the webservice name and should be the serviceURI portion of the URL used by client applications to invoke the Web Service.</p> <p>Now, the URI in the web-services.xml file generated from wsdl2service uses the serviceURI portion of the full URL as in:</p> <p>http(s)://host:port/contextURI/serviceURI</p> <p>If the full URL to invoke the Web Service is not in the above format, the URI must be correctly entered in the web-services.xml file manually.</p>
CR192510	<p>When WSSEClientHandler was used in the webllogic webservice client to invoke a secure non-WebLogic Web Service, the client got a SOAPException: unknown source type, when it processed the response.</p> <p>Now, under these circumstances, no exception is thrown.</p>
CR188548	<p>When clientgen created [WebserviceName].XML, if the web service class name contained multibyte encoding, the XML was created by system default encoding and did not specify its encoding in the file.</p> <p>Weblogic Server now Writes out the header and content with the specified encoding (-Dwebllogic.webservice.i18n.charset) or UTF-8 by default.</p>
CR190527	<p>When Message Level Security was enabled for Web Services, if the request message exceeded a certain size, WebLogic Server was throwing the JSAFE_InvalidUseException when trying to encrypt the request message.</p> <p>WebLogic Server no longer throws the JSAFE_InvalidUseException when trying to encrypt a large request message.</p>
CR183382	<p>When QName Type was used as an attribute inside a complex type, it was not marshalled properly.</p> <p>Now, WebLogic Server marshalls the QName Type when it is used as an attribute inside a complex type.</p>
CR172119	<p>WebLogic Server does not support overloading operations in Web Services Description Language (WSDL) due to a SOAP limitation. However, WebLogic Server was not detecting these operations during clientgen which was then failing with compilation errors.</p> <p>Now, for overloaded WSDL operations, clientgen throws the WSDLParseException instead of resulting in compiler errors.</p>

Change Request Number	Description
CR161835	<p>Using reliable Web Services with JMS Transport was generating an AssertionError. Now, WebLogic Server supports JMS Transport with reliable Web Services.</p>
CR190030	<p>When running clientgen, codegen was invoking the protected constructor, Enumeration, and mapping SOAPElement did not propagate to child when the super was SimpleType. As a result, clientgen was failing with two compilation errors.</p> <p>WebLogic Server no longer tries to invoke the Enumeration constructor and mapping of SOAPElement now propagates to child when the super is SimpleType. Thus, clientgen is now compiling as expected.</p>
CR189853	<p>The fillDetail method of weblogic81.webservice.util.FaultUtil uses the code: Throwable.getCause which is a JDK 1.4 API. In JDK 1.3, the getCause method is not present in Throwable.</p> <p>As a result, when using a WebLogic Server 6.1 client to access a WebLogic Server 8.1 SP3 (or earlier) webservice over HTTPS, the following error occurred:</p> <pre>java.lang.NoSuchMethodError at weblogic81.webservice.util.FaultUtil.fillDetail(FaultUtil.java:84)</pre> <p>Catching NoSuchMethodError fixes the problem, thus eliminating the error.</p>
CR187327	<p>Using clientgen to generate a client jar against a WSDL of a web service generated from WebLogic Workshop 8.1.2 the following failure occurred;</p> <pre>weblogic.xml.schema.binding.internal.builtin.XSDDurationSerializer; overridden method does not throw weblogic.xml.schema.binding.SerializationException</pre> <p>WebLogic Server now allows the use of clientgen to generate such a client jar.</p>
CR186319	<p>When the number of contiguous white spaces in an xml file exceeded 32, WebLogic Server required a long time to parse the file.</p> <p>WebLogic Server now has a larger buffer size and parses xml files more efficiently.</p>
CR182483	<p>WebLogic Server did not encode invalid XML characters via character references in the response.</p> <p>WebLogic Server now encodes invalid XML characters via character references.</p>

Change Request Number	Description
CR181987	<p>There were two issues:</p> <ol style="list-style-type: none"> <li>1. The first message in the JMS queue failed to deliver after undeployment of an application.</li> <li>2. WebLogic Server created the WebServiceOperationRuntime mbean twice when two ports pointed to the same operation and caused a <code>javax.management.InstanceAlreadyExistsException</code>.</li> </ol> <p>The first message is now delivered when an application is undeployed.</p> <p>WebLogic Server no longer creates the WebServiceOperationRuntime mbean twice even when two ports point to the same operation.</p>
CR181775	<p>WebLogic Server generated the type dynamically when <code>xmlType</code> was not an element of the known schema during deployment, hence the element names from the original imported schema were not preserved in the generated wsdl.</p> <p>A system property <code>-Dweblogic.webservice.noDynamicTypeGeneration</code>, has been added to webLogic Server, which when set to true, will preserve the original element names.</p>
CR180643 CR200646	<p>WebLogic Server was experiencing a memory leak of <code>GenericClassLoader</code> while performing the following steps:</p> <ol style="list-style-type: none"> <li>1. deploy a webservice application</li> <li>2. invoke the webservice</li> <li>3. undeploy the webservice application</li> </ol> <p>WebLogic Server now has no memory leaks associated with <code>GenericClassLoader</code>.</p>
CR178683	<p>Synchronization was added to <code>XMLSignature.validate()</code> in 8.1sp3 to avoid multi-threading issue in the xml message signature validation process. This had an adverse effect on performance.</p> <p>WebLogic Server has now eliminated the multi-threading issue in the xml message signature validation process and removed the synchronization to increase performance.</p>
CR177611	<p>There was a namespace problem in <code>web-service.xml</code> when <code>servicegen</code> used the type mapping file generated by <code>autotype</code>.</p> <p><code>servicegen</code> now recognizes namespaces in the type mapping file generated by <code>autotype</code>.</p>
CR177273	<p>The test page could not accept empty input for datatypes other than <code>java.lang.String</code>.</p> <p>WebLogic Server can now accept empty input for all built-in Java types on the test page.</p>

Change Request Number	Description
CR176392	<p>When trying to use WebLogic Server 6.1 based Java components to call WebLogic Server 8.1 services, an error occurred while attempting to invoke <code>weblogic.webservice.WebServiceLogger</code>.</p> <p><code>wsclient81.jar</code> now contains the <code>weblogic.webservice.WebServiceLogger.class</code>, thus eliminating this error.</p>
CR172463	<p>WebLogic Server only passed JSESSIONID cookies from the webservice client to the server. WebLogic Server now supports the use of non-JSESSIONID cookies.</p>
CR136707 CR196499	<p>WebLogic Server could not get <code>HttpServletRequest</code> inside a webservice, and thus could not retrieve a remote client ip.</p> <p>WebLogic Server now provides a public API to retrieve <code>HttpServletRequest</code> inside a webservice via <code>WebServiceSession.getRequest()</code>. It must be cast to <code>HttpServletRequest</code>.</p> <p>Now one can retrieve <code>HttpServletRequest</code> in this manner:</p> <pre>WebServiceContext wsContext = WebServiceContext.currentContext(); WebServiceSession vHttp = wsContext.getSession(); String remoteAddr = ((HttpServletRequest)(vHttp.getRequest())).getRemoteAddr();</pre>
CR183555 CR192555 CR192568	<p>JAX-RPC clients did not maintain session stickiness using the <code>SESSION_MANTAIN_PROPERTY</code>.</p> <p>WebLogic Server now supports session stickiness if <code>SESSION_MANTAIN_PROPERTY</code> is set as follows:</p> <p>On the Call object:</p> <pre>call.setProperty(Call.SESSION_MANTAIN_PROPERTY, "true"); or call.setProperty(Call.SESSION_MANTAIN_PROPERTY, new Boolean("true"));</pre> <p>On the Stub object:</p> <pre>(Stub)port)._setProperty(Stub.SESSION_MANTAIN_PROPERTY, "true"); or (Stub)port)._setProperty(Stub.SESSION_MANTAIN_PROPERTY, new Boolean("true"));</pre>

Change Request Number	Description
CR183676 CR195447 CR128604	<p>When the complex type contained a sequence and an element was defined with <code>maxOccurs&gt;1</code> in the sequence, WebLogic Server tried to add an unnecessary outer layer for the array of elements in the response, which resulted in a <code>weblogic.xml.stream.XMLStreamException: attempt to add an attribute to a null start element. exception being thrown.</code></p> <p>Additional condition checking was added to WebLogic Server so that does not add an unnecessary outer layer for the array of elements.</p>
CR185260 CR196682	<p>Autotype generated flawed Java Bean representations of XML Schema <code>complexType</code>s when those types contained both attributes and elements in a choice block. Bean setter functions for the elements of the choice block cleared the values of any attributes associated with the bean.</p> <p>Code was added to prevent clearing the attribute values and WebLogic Server now generates correct representations of XML Schema <code>complexType</code>s.</p>
CR185228	<p>The WebService SSL client failed to connect the service when <code>"weblogic.webservice.client.ssl.strictcertchecking"</code> was not set to false.</p> <p>WebLogic Server now connects to the service with this property set to either true or false.</p>

## WebLogic Tuxedo Connector

<b>Change Request Number</b>	<b>Description</b>
CR127288	A local domain making a CORBA request via WTC to a remote Tuxedo domain CORBA C++ server causes WTC to hang. WTC hangs because it ties up a WLS ExecuteThread at a CORBA request to Remote Tuxedo domain, where the target CORBA C++ server has crashed.
CR182219 CR201408	After configuring WTC, looking up the Tuxedo domain in the JNDI tree of the server causes a connection attempt to Tuxedo using the TGIOP protocol. This protocol is not in the protocol list maintained on the server, resulting in an UNKNOWN_QOS error. If the protocol is unknown then there is no attempt to get its QOS and the security check that checks if an attempt is being made to connect using a secure protocol to a non-secure port is skipped.
CR185475	Support XA transaction timeout for Oracle by setting <code>CALL_SET_TRANSACTION_TIMEOUT</code> to true.
CR185974	Transactional CORBA call to Tuxedo hangs indefinitely if Tuxedo CORBA server dies. The transaction stays in <code>GWTX_ACTIVE (TMGACTIVE)</code> across domains, and it ignores the <code>setTransactionTimeout()</code> .
CR186628	tBridge does not get cleaned when WTC service is undeployed. This leaks tBridge threads.
CR187267	Too many context switches occur on WTC when using a large number of threads.
CR187873	Forced shutdown during transactional activity causes error condition. WLS is unable to boot and this results in a memory leak.
CR189922	If the field of type <code>CARRAY</code> is empty (zero length), Tuxedo will not get zero length as intended, but receives 4 bytes.
CR189983	If a WTC service is undeployed and redeployed again, the tBridge redirection is not restarted.
CR194516	Under transaction, WLS execution thread hangs forever when a CORBA exception (due to ATMI error on Tuxedo CORBA server) is thrown back via WTC.
CR196598	An unknown WTC remote Tuxedo domain trying to connect to WTC will cause WTC memory leak.
CR198265	XA exceptions are not caught by WTC at the time of calling <code>xa_end</code> .

---

<b>Change Request Number</b>	<b>Description</b>
CR201408	Clone of CR182219.  After configuring WTC, looking up the Tuxedo domain in the JNDI tree of the server causes a connection attempt to Tuxedo using the TGIOP protocol. This protocol is not in the protocol list maintained on the server, resulting in an UNKNOWN_QOS error. If the protocol is unknown then there is no attempt to get its QOS and the security check that checks if an attempt is being made to connect using a secure protocol to a non-secure port is skipped.
CR201682	WTC export is uniquely identified by Local Access Point and Remote name, not Resource name.

---

## WebLogic Workshop

---

<b>Change Request Number</b>	<b>Description</b>
CR189420	<p>WebLogic Server was iterating over an array list, manipulating the list as it iterated, and then throwing a <code>ConcurrentModificationException</code>.</p> <p>WebLogic Server no longer has a synchronization problem and is therefore no longer throwing <code>ConcurrentModificationExceptions</code>.</p>

---

## XML

---

<b>Change Request Number</b>	<b>Description</b>
CR175990	<p>WebLogic Server's built-in xerces did not accept non-escaped multi-byte strings in the URL. WebLogic Server now escapes multi-byte strings based on UTF-8 character encoding.</p>
CR135726	<p>When WebLogic Server is unable to handle a given schema construct, it is mapped to <code>SOAPElement</code>.</p> <p>When this occurred, autotype gave no indication that it had encountered an un-supported schema construct and only examination of the generated classes showed that something had gone wrong.</p> <p>WebLogic Server now provides a warning message when it is unable to handle a given schema construct and falls back to <code>SOAPElement</code>.</p>
CR184002 CR198219	<p>When the <code>&lt;/soap:Envelope&gt;</code> tag was removed from the bottom of the soap-signed documents, and the certifier was run against it, it was still verified.</p> <p>WebLogic Server now throws an <code>XMLSignatureMalformedException</code> for documents that are not well formed.</p>

---

# Resolved Problems for Service Pack 2

Service Packs are cumulative; Service Pack 2 contains all the fixes made in earlier Service Packs released for WebLogic Server 8.1.

The following sections describe problems that were resolved for the release of WebLogic Server 8.1 Service Pack 2.

- [“Administration Console” on page 5-2](#)
- [“Classloader” on page 5-6](#)
- [“Cluster” on page 5-6](#)
- [“Connector” on page 5-8](#)
- [“CORE” on page 5-10](#)
- [“Deployment” on page 5-14](#)
- [“EJB” on page 5-17](#)
- [“Installation” on page 5-26](#)
- [“J2EE” on page 5-27](#)
- [“JCOM” on page 5-28](#)
- [“JDBC” on page 5-28](#)
- [“JMS” on page 5-33](#)

- [“JNDI” on page 5-38](#)
- [“JTA” on page 5-40](#)
- [“Node Manager” on page 5-42](#)
- [“Plug-Ins” on page 5-43](#)
- [“RMI/RMI-IIOP” on page 5-45](#)
- [“Security” on page 5-49](#)
- [“Servlets & JSPs” on page 5-54](#)
- [“System Administration” on page 5-69](#)
- [“Tools” on page 5-75](#)
- [“Web Services” on page 5-78](#)
- [“WTC” on page 5-88](#)
- [“XML” on page 5-91](#)

## Administration Console

Change Request Number	Description	Release Fixed
CR076953	<p>If you created a security role using the context-sensitive (right-click) menu item "Define Policies and Roles for individual beans..." and added the security role to the security policy (defined using the same context sensitive menu/form sequence), the desired WebLogic resource could not be used by an external JCom client.</p> <p>The context-sensitive menu items create a scoped role, and clients outside the scoped role could not use the WebLogic resource.</p> <p>Adding a "Define JCom Roles" menu item to the JCom node solved this problem by allowing revision of the scope to include the current client.</p>	7.0 SP3 8.1 SP2
CR078764	<p>Security settings in the Administration Console for Web services did not function properly.</p> <p>A code change removed the ability to set policy and roles for Web services in the Administration Console.</p>	7.0 SP3 8.1 SP2

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR081673	Realms created by the Administration Console lacked the <code>UserLockoutManager</code> on their <code>RealmMBeans</code> . The Administration Console now creates a <code>ULMbean</code> when it creates a new realm.	7.0 SP3 8.1 SP2
CR082335	The cursors created by the Administration Console for listing users and groups caused potential memory leaks when they did not close. Following a code fix, the cursors close.	7.0 SP3 8.1 SP2
CR091853	For a three-node cluster on separate Solaris 2.8 machines, the Administration Server threw an <code>InstanceNotFound</code> exception when accessing the default <code>Execute Queue</code> . The problem has been resolved.	7.0 SP4 8.1 SP2
CR099721	After edits to <code>Cookie Max Age Secs</code> using the Administration Console, the value sometimes reverted to the value before the edit. A code change removing a lower limit to the value fixed the problem.	7.0 SP3 8.1 SP2

## Resolved Problems for Service Pack 2

Change Request Number	Description	Release Fixed
CR100006	<p>WebLogic Server sometimes displayed the following <code>NullPointerException</code> when a domain contained two Administration Servers and you tried to access the servers via the WebLogic Server Administration Console:</p> <pre>java.lang.NullPointerException at weblogic.management.console.helpers.UrlHelper.buildIconList( UrlHelper.java:149) at weblogic.management.console.helpers.UrlHelper.getIcon(UrlHel per.java:174) at weblogic.management.console.tags.SmartNavNodeTag.getIcon(Sma rtNavNodeTag.jacva:111) at weblogic.management.console.tags.SmartNavNodeTag.inferStuffF romContext(SmartNavNodeTag.java:96) at weblogic.management.console.tags.SmartNavNodeTag.doStartTag( SmartNavNodeTag.java:44) at weblogic.management.console.webapp.__domain.__nav.__jspService (__nav.java:301) at weblogic.servlet.jsp.JspBase.service(JspBase.java:27) at weblogic.servlet.internal.ServletStubImpl.invokeServlet(Serv letStubImpl.java:262) at weblogic.servlet.internal.ServletStubImpl.invokeServlet(Serv letStubImpl.java:198) at weblogic.servlet.internal.RequestDispatcherImpl.forward(Requ estDispatcherImpl.java:250) [...]</pre> <p>This problem was solved with a code fix.</p>	6.1 SP6 8.1 SP2
CR103104	<p>The <code>UserReader</code> MBean did not display users in the Administration Console. An Authentication provider can read users in the Administration Console while implementing the <code>UserEditor</code> but with <code>UserReader</code>, the message when trying to display users is "There are no Authentication providers available that support the creation of Users." The same thing was happening for Groups with the <code>GroupEditor</code> and <code>GroupReader</code> interfaces.</p> <p>The Administration Console was still calling the <code>userEditor.createUser</code> method, and never getting to the <code>UserReader.listUsers</code> method.</p> <p>The MBean has been fixed to look for appropriate provider type.</p>	7.0 SP3 8.1 SP2
CR105598	<p>New testing with an Active Directory LDAP uncovered a problem where a call to <code>group.isMember()</code> would fail if a member name contained a comma character (,).</p> <p>This problem was solved with a code fix.</p>	6.1 SP6 8.1 SP2

Change Request Number	Description	Release Fixed
CR106027	Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03-28.01.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03-28.01.jsp</a> .	7.0 SP3 8.1 SP2
CR111321	<p>If there is only one server in a WebLogic Server domain, the WebLogic Server Administration Console does not display virtual hosts as potential targets during the deployment of a Web application. This was done to save users the extra step of having to select a target when there is only one target. However, this prevented users from deploying a Web application to a virtual host directly. Users could only deploy the Web application to the server and then re-target it to a virtual host. This worked fine for the first Web application, but for the second Web application, an exception was thrown indicating that the context was already in use.</p> <p>For Web applications, the Administration Console now checks for any virtual hosts configured in the domain and displays them as potential targets. As a result of this fix, customers should now be able to target Web applications to a virtual host directly from the Administration Console. If there are no virtual hosts, the Administration Console assumes that the lone server is the target for the Web application, just as it did prior to this fix.</p>	8.1 SP2
CR112726	<p>In the <code>FileChooserControlTag</code>, WebLogic Server defaulted to UTF-8 encoding for path settings.</p> <p>This fix causes WebLogic Server to extract the correct charset from the catalog and use that instead of defaulting to UTF-8. As a result of this change, paths are now displayed correctly (without any garbled characters).</p>	8.1 SP2
CR120537	<p>Users of the WebLogic Server Administration Console could not delete foreign JMS destinations.</p> <p>A trash can icon (delete) has been added to the foreign JMS destinations table. Users can now delete foreign JMS destinations.</p>	8.1 SP2
CR120603	<p>Editing the load order in the WebLogic Server Administration Console caused spaces to be embedded in the <code>path</code> attribute of the <code>config.xml</code> file's <code>&lt;Application&gt;</code> tag.</p> <p>This problem occurred because the method <code>pathToFormValue</code>, which is used to breakdown long path strings to more manageable pieces, was rejoining the strings with extra spaces.</p> <p>The code was changed to remove these extra spaces. As a result of this fix, the extra spaces in the <code>path</code> attribute will no longer be inserted when the <code>config.xml</code> is updated.</p>	8.1 SP2
CR124344	Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03_41.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03_41.00.jsp</a> .	8.1 SP2

## Classloader

Change Request Number	Description	Release Fixed
CR124348	<p>Client programs could not use <code>java.lang.reflect.Proxy</code> to access proxy objects deployed in WebLogic Server, unless the proxy classes were added to the system classpath. If an object did not reside in the system classpath, the client would receive a <code>ClassNotFoundException</code>.</p> <p>The <code>resolveProxyClass()</code> method was implemented to load interfaces from the application-specific classloader as well as the system classloader.</p>	<p>6.1 SP6</p> <p>8.1 SP2</p>

## Cluster

Change Request Number	Description	Release Fixed
CR094837	<p>The following sequence was causing a <code>ClassCastException</code>: define a server, assign that server to a cluster, create a Unix machine with the same name as the server, assign them to each other, stop and restart the Administration Server, and select the server name from the cluster or from the Servers list. Trying to bring up the Managed Server also caused a <code>ClassCastException</code>.</p> <p>Fixing a problem in a process that resolves an <code>UnresolvedMBean</code> during startup solved this problem.</p>	<p>7.0 SP3</p> <p>8.1 SP2</p>
CR105698	<p>If two clusters have between them two distributed queues with the same JNDI name, and a message-driven bean is deployed on both the clusters, messages were being consumed in only one of the clusters.</p> <p>A message-driven bean now goes through the entire collection of distributed queues in a domain and adds their members appropriately.</p>	<p>7.0 SP3</p> <p>8.1 SP2</p>

Change Request Number	Description	Release Fixed
CR108127	<p>An error occurred when trying to update a secondary session. Given three clustered server instances: Server A, Server B, and Server C.</p> <ol style="list-style-type: none"> <li>1. Initiate a session on Server A, PRIMARY A, SECONDARY C</li> <li>2. Load balancing hardware migrates the session to Server C, PRIMARY C, SECONDARY B</li> <li>3. Load balancing hardware migrates the session to Server A again</li> </ol> <p>This error occurred on Server A:</p> <pre>&lt;Jun 4, 2003 4:24:56 PM PDT&gt; &lt;Debug&gt; &lt;Cluster&gt; &lt;Error updating secondary for 3535724191309537720 on 7312270160516507840S:&lt;IPaddressdeleted&gt;: [7005,7005,7002,7002,7005,7002,-1]:&lt;IPaddressdeleted&gt;:7005, &lt;IPaddressdeleted&gt;:7005,&lt;IPaddressdeleted&gt;:7005:mydomain:mserver3. Re-creating secondary.&gt;</pre> <p>This error occurred on Server C:</p> <pre>&lt;Jun 4, 2003 4:24:56 PM PDT&gt; &lt;Info&gt; &lt;Cluster&gt; &lt;Lost 0 replication updates of object 3535724191309537720. Re-fetching secondary.&gt;</pre> <p>The problem was solved by a code change.</p>	6.1 SP6 8.1 SP2
CR108893	<p>The <code>ClusterRuntime.getName()</code> method returned the name of a server in the cluster, rather than the name of the cluster.</p> <p>This method has been fixed to properly return the name of the cluster.</p>	8.1 SP2
CR127643	<p>When a dynamic proxy that implemented interfaces declared inside a Web application was put into the <code>HttpSession</code> and the session was replicable, WebLogic Server was not able to load the interface classes on the secondary server.</p> <p>To correctly resolve the dynamic proxy, the secondary server needs the name of the application where the interface resides. An <code>annotateProxyClass</code> was implemented in <code>MsgAbbrevOutputStream</code> to write the <code>applicationName</code> in the stream. On the receiving side, <code>resolveProxyClass</code> uses this application name to load the interface classes from the application.</p> <p>As a result of this change, dynamic proxies (implementing interfaces stored in the application archive) can be put into the <code>HttpSession</code> and be correctly replicated. There should be no side effects.</p>	8.1 SP2

## Connector

Change Request Number	Description	Release Fixed
CR091818	<p>There was no configuration parameter to disable Connection Proxy creation.</p> <p>This problem was resolved by adding a property called <code>use-connection-proxies</code> in the <code>weblogic-ra.xml</code> deployment descriptor.</p> <p>If the property is set to “true,” then proxies will be used without performing the test for connection proxy viability. That is, the case where a resource adapter attempts to cast the wrapped connection (proxy) that WebLogic Server generates, and the cast fails. If the property is set to “false,” then proxies will not be used. If no value is specified for the property, previous functionality will be used. (That is, the test for connection proxy viability will be performed and proxies will be used if and only if the test passes.)</p> <p>The only caveat to a late transaction enlistment is that client code cannot first acquire a connection handle, then start a transaction and have that connection automatically enlisted. If the client starts a transaction first and then acquires a connection handle, everything works fine. The absence of late-transaction enlistment implies nothing about the use of bean-demarcated vs. container-managed transactions. Both will work.</p>	8.1 SP2
CR105247	<p>When trying to deploy a RAR component with a set of its utility classes packaged as JAR files, WebLogic Server was unable to load the properties file, which is included in the JAR file.</p> <p>There was a "/" missing from the URL and this resulted in not being able to find the properties file. The code was modified to add the "/" when needed.</p>	8.1 SP2
CR106388	<p>During deployment of a RAR that uses the <code>ra-link-ref</code> element, called a 'logical' RAR, the logical RAR deployment was not getting the correct pool name or max-capacity as specified in its <code>weblogic-ra.xml</code> deployment descriptor. This caused unexpected behavior in WebLogic Integration.</p> <p>The element now functions properly.</p>	7.0 SP3 8.1 SP2

Change Request Number	Description	Release Fixed
CR106663	<p>New validation checks created incompatibility with previous versions.</p> <p>Validation checks were changed to be backwards compatible. Entries that should be nullable are now allowed. The following parameters continue to prohibit <code>null</code> values:</p> <ul style="list-style-type: none"> <li>• <code>connection-factory-name</code></li> <li>• <code>jndi-name</code></li> <li>• <code>shrinking-enabled</code> (true/false only)</li> <li>• <code>connection-profiling-enabled</code> (true/false only)</li> <li>• <code>match-connections-supported</code></li> <li>• <code>logging-enabled</code> (true/false only)</li> <li>• <code>map-config-property-name</code></li> <li>• <code>initiating-principal resource-principal</code></li> </ul> <p>Also, <code>log-filename</code> does not allow <code>null</code> if <code>logging-enabled</code> is <code>true</code>.</p> <p>Because of incompatible validation checks, some adapters that would deploy without error in 7.0 may not deploy in 8.1 or 8.1 SP1, because of <code>null</code> or empty values in the <code>weblogic-ra.xml</code> descriptor. The validation checks have been modified as of 8.1 SP2 so that it should now be backwards compatible with all previous versions.</p>	8.1 SP2
CR106960	<p>When a RAR was part of an EAR, redeployment of the EAR failed because the Connector Module code did not handle redeployment properly.</p> <p>Fixing the redeploy logic in the Connector module resolved the problem.</p>	7.0 SP4 8.1 SP2
CR111229	<p>In WebLogic Server 8.1 SP1, with multiple clients using connections for an adapter, XAER_PROTO errors could occur while initializing a connection to be used in a transaction. The problem was caused by the fact that a connection and the state of its transaction was not always being completely resolved before releasing the connection back to the pool of available connections. This could allow another client to start using the connection before its state was completely reset by the previously calling thread.</p> <p>In WebLogic Server 8.1SP2, the server ensures that the connection and transaction state are completely resolved before releasing the connection back to the pool of available connections. Thus, connection requests will no longer create XAER_PROTO exceptions while initializing the connection to be used in a transaction.</p>	8.1 SP2

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR112162	<p>If a resource adapter's implementation of <code>ManagedConnectionFactory.matchConnections(connectionSet)</code> sent the <code>CONNECTION_ERROR_OCCURRED</code> event for a <code>ManagedConnection</code> that was in the <code>connectionSet</code>, then the <code>ManagedConnection</code> instance would be destroyed by the Connector container calling <code>ManagedConnection.destroy()</code>, but it would not be properly removed from the pool of available managed connections.</p> <p>This problem occurred because during the processing of the error event, the code was only looking in the set of reserved connections to perform the removal from the pool. In other words, when an error occurred, the Connector container assumed that the connection was currently in use.</p> <p>During the processing of the <code>CONNECTION_ERROR_OCCURRED</code> event, the code will now check the set of available connections if the destroyed connection is not found in the set of reserved connections. With the fix, the <code>ManagedConnection</code> will be destroyed and removed from the pool appropriately when a <code>CONNECTION_ERROR_OCCURRED</code> event is invoked.</p>	8.1 SP2

## CORE

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR092375	<p>Manually doing a lookup and then persisting and caching the handle did not work when the methods were conversational.</p> <p>Changes to serialization and deserialization of the <code>EJBHandle</code> object resolved the problem.</p>	7.0 SP2 8.1 SP2
CR096091	<p>When starting WebLogic Server as an Windows service using a classpath from a file, the error "Thread created successfully! Exception in thread "main" java.lang.NoClassDefFoundError: weblogic/Server" was thrown if classpath file is over about 2k length.</p> <p>The problem was solved by correcting an error related to reading the classpath from a file.</p>	6.1 SP6 8.1 SP2

---

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR102848	The CoreHealthMonitor was holding the ExecuteThreadManager lock while performing significant work, resulting in deadlocks and Administration Console freezes. The code was changed so that CoreHealthMonitor now uses ExecuteThreadRuntimeMBean to get a list of stuck threads and avoids logging from all places in the ExecuteThreadManager if ETM lock is held.	7.0 SP3 8.1 SP2

---

Change Request Number	Description	Release Fixed
CR105516	<p data-bbox="229 418 971 440">Stateful session EJB failover did not work when multiple failovers were required.</p> <p data-bbox="229 461 1065 574">In a three node cluster a JSP creates or calls a replica-aware stateful session EJB. The remote EJB stub is stored in the http session. After each call to the stateful session EJB the updated EJB stub is also updated in the HTTPsession to reflect any changes for the EJB replica list (primary/secondary).</p> <p data-bbox="229 591 998 644">The following call sequence with the same browser window leads to a <code>java.rmi.ConnectException</code> when only one node of the cluster survives:</p> <ol data-bbox="229 656 1065 956" style="list-style-type: none"> <li data-bbox="229 656 592 678">1. All three cluster nodes are running.</li> <li data-bbox="229 696 1038 748">2. Making a call to node1, create EJB and store remote in HTTP session (HTTP session replication is enabled)</li> <li data-bbox="229 765 360 788">3. Kill node1</li> <li data-bbox="229 805 1065 874">4. Make a call to the secondary node2, the EJB remote is retrieved from the replicated HTTP session and the call to the EJB works fine. After this EJB call again the remote is stored in the HTTP session.</li> <li data-bbox="229 892 360 914">5. Kill node2</li> <li data-bbox="229 932 827 954">6. Make a call to node3, get the EJB remote from HTTP session.</li> </ol> <p data-bbox="229 973 1065 1025">WebLogic Server tries to lookup the EJB on node2 and does not try to use node3 (this should now be the new secondary). The following exception is thrown on node3:</p> <pre data-bbox="229 1043 1065 1489"> java.rmi.ConnectException: Could not establish a connection with -3088833905169218734S:172.23.64.38:[7001,7001,7002,7002 ,7001,7002,-1]:mydomain:managed2, java.rmi.ConnectException: Destination unreachable; nested exception is: java.net.ConnectException: Connection refused: connect; No available router to destination at weblogic.rjvm.RJVMImpl.getOutputStream(RJVMImpl.java:275) at weblogic.rjvm.RJVMImpl.getRequestStream(RJVMImpl.java:408) at weblogic.rmi.internal.BasicRemoteRef.getOutboundRequest(Basi cRemoteRef.java:97) at weblogic.rmi.cluster.ReplicaAwareRemoteRef.invoke(ReplicaAwa reRemoteRef.java:255) </pre> <p data-bbox="229 1506 592 1529">The problem was solved with a code fix.</p>	6.1 SP6 8.1 SP2

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR105980	<p>A memory leak occurred when creating a <code>QueueReceiver</code> for each message on a Distributed Queue and persisted even after the <code>QueueReceiver</code> had been shut down.</p> <p>A code change corrected memory leaks in <code>BasicServiceOffer</code> and <code>DistributedDestinationImpl</code> objects.</p>	<p>7.0 SP3</p> <p>8.1 SP2</p>
CR110028	<p>The functionality to detect stuck threads in user-configured thread queues was not complete in WebLogic Server 7.0 and its service packs.</p> <p>Thread queues have been reworked and provisions made to identify an application thread from an internal server thread queue. Logic was added to loop through all application thread queues, checking for stuck threads. All application thread queues will now be monitored by the Core Health Monitoring Thread and proper warnings logged.</p>	8.1 SP2
CR110367	<p>User code such as servlet <code>destroy()</code> or <code>ejbRemove()</code> gets executed during shutdown. This fix ensures that lower order services like JMS and JDBC are available when this user code executes.</p>	8.1 SP2
CR110887	<p>When nodes in a cluster tried to get session information, they were attempting a lookup from a remote server, hence making a remote call that blocked a thread on the local server. When all nodes were doing the lookup under load, the execute thread queues became blocked and no server was in a position to send a response to the remote call (as all local threads were blocked).</p> <p>WebLogic Server now creates the stubs locally to avoid a remote call, until the point where it is necessary. When WebLogic Server makes the remote call, it lands on a separate replication queue on the remote server and the process does not block there.</p> <p>The replication behavior is still unchanged. As a result of this change, WebLogic Server only ensures less contention, avoids a deadlock situation, and reduces the number of remote calls.</p>	8.1 SP2
CR110892	<p>Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03-35.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03-35.jsp</a>.</p>	<p>6.1 SP6</p> <p>8.1 SP2</p>
CR120841	<p>WebLogic Server did not collect timer threads during orderly shutdown. This did not result in any observable problems.</p> <p>Shutting down the ORB now shuts down heartbeat threads. This change should not result in any changed behavior.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR123846	<p>ClassCastExceptions were being thrown when a transaction was started in a user thread.</p> <p>This problem occurred because the parallel XA implementation was using a Kernel method, which was assuming that all requests would be executed on a ExecuteThread, and this method was casting the current thread as an ExecuteThread. When the work ran in a user thread on the server side, the exception would be thrown.</p> <p>This problem was resolved by allowing the execution to happen on any thread.</p>	8.1 SP2

## Deployment

Change Request Number	Description	Release Fixed
CR101760	<p>Deleting a Web application from a Managed Server that had been stopped resulted in errors similar to the following:</p> <pre>&lt;Warning&gt; &lt;Management&gt; &lt;149311&gt; &lt;Rejecting deployment operations to non-running server, mgd1&gt; &lt;Warning&gt; &lt;Deployer&gt; &lt;149004&gt; &lt;Failures detected initiating weblogic.management.ManagementException: Rejecting deployment operations to non-running server, mgd1 task for application Remove application MyWebApp on mgd1&gt;</pre> <p>The Web application was not deleted.</p> <p>A code change fixed the problem by eliminating dangling WebAppComponentMBean references from the WebServer and VirtualHost MBeans.</p>	7.0 SP3 8.1 SP2
CR102928	<p>Web applications in an EAR deployed to a VirtualHost did not redeploy on domain restart.</p> <p>A code change causes applications that target a Virtual Host to restart when the domain restarts.</p>	7.0 SP3 8.1 SP2

Change Request Number	Description	Release Fixed
CR109668	<p>Running the following command caused all the modules in the application to transition from an active state to an inactive state, even though it was expected that only the specified module in the application would be undeployed.</p> <pre>java weblogic.Deployer -url &lt;admin_url&gt; -username &lt;user_name&gt; -password &lt;passwd&gt; -name &lt;app_name&gt; -undeploy -targets &lt;module_name&gt;@&lt;cluster_name&gt;</pre>	8.1 SP2
	<p>This problem occurred because in the <code>SlaveDeployer</code>, WebLogic Server did not factor in the possibility that a component could have been targeted to a cluster, and was performing checks based only on the server target type.</p> <p>The code has been modified to add a check for the possibility of a component target being of type cluster. Partial undeployment should now work as expected.</p>	
CR111065	<p>In certain instances, the DRS system is creating the delta list of deployment tasks for the <code>SlaveDeployer</code> initialization.</p> <p>A code fix ensures that DRS always uses the <code>MasterDeployer</code> for creating the deployment task deltas.</p>	8.1 SP2
CR113178	<p>In the <code>SlaveDeployer</code> code, the applications are sorted and added to a <code>HashSet</code> (<code>ApplicationsSet</code>). The <code>Iterator</code> taken out of this <code>HashSet</code> does not retain the order of the elements that are added to it. A <code>TreeSet</code> was used to store these elements in a sorted order, but it could not store duplicate applications with the same deployment order. (One will override the other.)</p> <p>Applications are now added to a <code>List</code> instead of a <code>Set</code>. As a result of this change, the <code>Iterator</code> preserves the order, and applications will be deployed per the <code>loadOrder</code> attribute of the <code>config.xml</code> file.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR120714	<p>The WLDeploy task was printing the password to stdout. For example, if build.xml contained:</p> <pre data-bbox="229 491 1056 739"> &lt;project name="cr120714" basedir="."&gt; &lt;taskdef name="wldeploy" classname="weblogic.ant.taskdefs.management.WLDeploy" /&gt; &lt;target name="deploy-app" &gt; &lt;wldeploy action="redeploy" nostage="true" name="CR120714App" user = "system" password = "gumby1234" adminurl = "t3://172.17.24.109:7001" verbose = "false" debug = "false" /&gt; &lt;/target&gt; &lt;/project&gt; </pre> <p>the output of ant deploy-app was:</p> <pre data-bbox="229 795 1056 1107"> Buildfile: build.xml deploy-app: [wldeploy] weblogic.Deployer -nostage -noexit -name CR120714App -adminurl t3://172.17.24.109:7001 -user system -password gumby1234 -redeploy [wldeploy] Initiated Task: [1] [Deployer:149026]Redeploy application CR120714App on myserver. [wldeploy] Task 1 completed: [Deployer:149026]Redeploy application CR120714App on myserver. [wldeploy] Deployment completed on Server myserver [wldeploy] </pre> <p>The original password is now replaced with "*****" before printing the command to stdout.</p>	8.1 SP2

# EJB

Change Request Number	Description	Release Fixed
CR087151	<p>For a stateless session bean, when the <code>weblogic-ejb-jar.xml</code> contained:</p> <pre>&lt;stateless-bean-is-clusterable&gt;False&lt;/stateless-bean-is-clusterable&gt;</pre> <p>and the bean was deployed on a cluster, this error was generated:</p> <pre>&lt;Mar 14, 2003 3:35:00 PM PST&gt; &lt;Error&gt; &lt;Cluster&gt; &lt;Conflict start: You tried to bind an object under the name ejb20-statelessSession-TraderHome_EO in the JNDI tree. The object you have bound from 172.17.24.112 is non clusterable and you have tried to bind more than once from two or more servers. Such objects can only deployed from one server.&gt;</pre> <p>This problem was resolved by a code fix to ensure that JNDI bindings for non-clustered stubs are not replicated.</p>	6.1 SP5 8.1 SP2
CR094073	<p>Before running a finder method on the BMP beans, the EJB container was not synchronizing the bean state with the database.</p> <p>BMP entity bean handling has been changed to conform to the EJB 2.0 specification. Per the EJB specification, the EJB container now synchronizes the entity bean state by invoking <code>EJBStore</code> before running a Finder method. This does not affect the <code>findByPrimaryKey</code>.</p> <p>In a transaction, if a finder method is invoked on a BMP bean, it will now trigger the <code>ejbStore</code> of all modified beans (BMP and CMP beans) participating in the transaction. Thus, BMP beans might get more <code>ejbStore</code> callbacks within a transaction.</p>	7.0 SP3 8.1 SP2
CR095173	<p>The <code>idle-timeout-seconds</code> element determined how long the EJB container waits before passivating stateful session beans, that is, removing them from cache and writing them to disk. The EJB container also used this element to determine how long to wait before removing passivated EJBs from the disk. However, some customers wanted stateful session beans to remain on disk longer than <code>idle-timeout-seconds</code>. They wanted to specify how long stateful session beans stay idle in the cache and how long they stay idle on disk using two different elements.</p> <p>This enhancement is made possible by introducing the new element <code>session-timeout-seconds</code>, which specifies how long the EJB container waits before removing an idle stateful session bean from disk.</p>	7.0 SP3 8.1 SP2

Resolved Problems for Service Pack 2

Change Request Number	Description	Release Fixed
CR096182	Dynamic EJB-QL incorrectly passed long arguments as MAX_INT to the underlying SQL, resulting in an unsuccessful search. A code fix resolved the problem.	7.0 SP3 8.1 SP2
CR096395	The container was failing to call unsetEntityContext when the entity bean was undeployed. An undeploy method was added to the BaseEntityManager to clean up the pool when an entity bean is undeployed. This resolved the problem.	7.0 SP3 8.1 SP2
CR097913	<p>Combining COUNT and EXISTS in a dynamic EJB-QL request resulted in invalid SQL and this exception:</p> <pre data-bbox="229 696 1056 861">[java] javax.ejb.FinderException: Exception in dynamicQuery while preparing or executing statement: 'weblogic.jdbc.rmi.SerialStatement@62b30' [java] java.sql.SQLException: ORA-00937: not a single-group group function [java] [java] java.sql.SQLException: ORA-00937: not a single-group group function</pre> <p>An unnecessary column was being added to the main select list while parsing a subquery. A fix to the parser code solved the problem.</p>	7.0 SP3 8.1 SP2
CR099626	<p>ejbc generated invalid SQL for an ejbSelect query. The query failed because table aliases were generated incorrectly, resulting in this error:</p> <pre data-bbox="229 1029 642 1046">ORA-00904: invalid column name</pre> <p>The problem was corrected with a code fix.</p>	6.1 SP6 8.1 SP2
CR099760	A field optimization was implemented for EJB 1.1 CMP beans, so that fields that are updated, but whose value did not change, are not written to back to the database. The optimization is only done for primitives and immutable objects.	6.1 SP6 8.1 SP2
CR100832	Column version numbers were not updating correctly with Optimistic concurrency enabled, because blind writes were not checked . The code was changed so that blind writes are checked with Optimistic concurrency.	7.0 SP3 8.1 SP2
CR102481	<p>In some bean-managed stateful session beans, an IllegalStateException occurred with bean-demarcated transactions when the number of bean instances is greater than max-beans-in-cache when the cache is full and activations and passivations occur.</p> <p>A code change results in the creation of one replacer per passivation request, so that each thread has its own replacer.</p>	7.0 SP3 8.1 SP2

Change Request Number	Description	Release Fixed
CR103047	<p>It was reported that transaction timeouts on MDBs were not logged. When the time an MDB takes to process a message from a JMS Destination exceeds the transaction timeout limit, the transaction is rolled-back and the message is place backed on the destination for re-delivery, but no transaction-timeout or transaction-rollback messages were logged.</p> <p>This problem was resolved by a code change to the MDListener code to report the transaction timeouts.</p>	6.1 SP6 8.1 SP2
CR103391	<p>The JDBC driver from Microsoft has a limitation whereby for a given result set of rows and columns, the <code>getXXX</code> method can only be called once per row. This limitation applied only if the query columns included a text or image column. WebLogic Server-generated JDBC obtained the key value from a row—for example, using a <code>getLong(1)</code>—and then passed the result set to another routine that read all the column values, including the first, re-executing a <code>getLong(1)</code>. This re-execution caused the Microsoft driver to throw an exception.</p> <p>The problem is resolved by a code fix that avoids parsing the primary key columns in the <code>resultSet</code> twice.</p>	7.0 SP4 8.1 SP2
CR105857	<p>EJBs with Bean-Managed Persistence and with concurrency set to exclusive called <code>ejbFindByPrimaryKey</code> rather than finding the bean in the cache.</p> <p>A code change resolved the problem so that the EJB finds the bean in the cache.</p>	7.0 SP3 8.1 SP2
CR106041	<p><code>ejbc</code> now runs a compliance check that disallows optimistic concurrency for BMP (bean-managed persistence) beans.</p> <p>Optimistic concurrency is a feature of CMP (container-managed persistence) beans in EJB 2.0. See <a href="#">“Choosing a Concurrency Strategy”</a> in <i>Programming WebLogic Enterprise JavaBeans</i>.</p>	7.0 SP4 8.1 SP2
CR106136	<p>Getter methods for a EJB 1.1 CMP bean did not call <code>isModified()</code> when <code>delay-updates-until-end-of-tx</code> was set to false.</p>	6.1 SP6
CR111025	<p>A session EJB called an entity EJB's getter methods. Both EJBs had container managed transaction with transaction attribute set to <code>Required</code>. Each call to a getter method is followed by a call to <code>ejbStore()</code> and <code>delay-updates-until-end-of-tx</code> was false. However, before calling <code>ejbStore</code> on the bean the container did not call the <code>isModified</code> method. The <code>isModified()</code> method was only called when the transaction committed.</p> <p><code>ejbStore</code> should be called from <code>postInvoke()</code> depending on the result of the <code>isModified</code> method in the bean. The problem was solved with a code fix.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR106711	<p>Optimistic concurrency with a version column did not work in combination with bulk updates. (An "ORA-01008: not all variables bound" SQLException was thrown.)</p> <p>This problem was resolved by using the correct bean state tracking variable so the EJB container sets all of the variables in the SQL statement, including the version column. Optimistic concurrency with a version column now works in combination with bulk updates.</p>	8.1 SP2
CR107447	<p>The EJB QL query generator could not handle:</p> <pre data-bbox="270 638 552 661">SELECT OBJECT(<i>target</i>)</pre> <p>where <i>target</i> is collection member variable as in:</p> <pre data-bbox="229 716 803 765">SELECT OBJECT(<i>target</i>) FROM EJBbean AS bean, IN(bean.collection) <i>target</i></pre> <p>This problem has been fixed. The EJB QL compiler now properly handles this case.</p>	8.1 SP2
CR107455	<p>The <code>ROManager</code> was forced to add duplicate entries to its internal LRU list. This resulted in unlimited growth of the list, which lead to <code>OutOfMemoryErrors</code>.</p> <p>The duplicate entries occurred because of an unnecessary call to the <code>initLastLoad()</code> method in the <code>DBManager</code> method <code>getBeanFromRS()</code>. Removing the unnecessary call fixed the problem.</p>	8.1 SP2
CR108456	<p>With a CMR one-to-many relationship, in a single transaction, if the bean on the one side was accessed but not changed, the dependency between the database operations was not resolved properly by the container. This resulted in a <code>java.sql.SQLException</code>.</p> <p>This problem was resolved by storing the related bean to the database before checking the state of the current bean.</p>	8.1 SP2
CR109621	<p>The EJB container automatically detects situations where an EJB has been changed in a possibly incompatible way since it was last compiled. If such a change is detected, the EJB is automatically recompiled. Previously, any change to a deployment descriptor would result in such recompilation. Some changes, however, are guaranteed not to change the EJB, such as adding whitespace to a deployment descriptor. Such a change should not have caused the EJB to be recompiled.</p> <p>The algorithm used to determine whether an EJB needs to be recompiled now disregards whitespace in deployment descriptors when determining whether the descriptor has changed. Thus, an EJB is not recompiled needlessly when only whitespace is changed in a deployment descriptor.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR110917	<p>When using stateful session beans under load with <code>home-is-clusterable</code> set to <code>true</code> and <code>replication-type</code> set to <code>InMemory</code>, the proxy objects on the secondary server that call the bean were not being unexported and were therefore leaking. This resulted in an <code>OutOfMemoryError</code>.</p> <p>WebLogic Server now unexports the proxy objects on the secondary server. When the Replicated bean receives a <code>becomeUnregistered()</code> callback from the <code>ReplicationService</code>, the corresponding proxy object on the secondary server is unexported in the <code>StatefulEJBHome.removeSecondary()</code> method.</p> <p>There should not be any negative side-effects on existing user applications.</p>	7.0 SP5 8.1 SP2
CR111233 CR122832	<p>When a MDB and JMS server were targeted to different WebLogic Server instances, the MDB was not receiving messages.</p> <p>This problem occurred because the EJB container was not creating the necessary MDB pool, as they are not co-located.</p> <p>This problem has been fixed. When the MDB's destination is non-distributed and non-migratable, then the co-location rule is not applied.</p>	8.1 SP2
CR111476	<p>If an EJB uses container-managed transactions, the EJB specification requires transaction attributes to be set for all EJB methods (excluding methods of <code>EJBObject\EJBLocalObject</code> for session beans and all methods of the <code>home\local-home</code> interface for session beans). Previously, a method not assigned a transaction attribute would use a default value of <code>Supports</code> for interface methods and <code>NotSupported</code> for a message-driven bean's <code>onMessage</code> method. These default values did not result in transactions being started and this caught several customers by surprise.</p> <p>If any EJB methods are missing a transaction attribute setting, the default transaction attribute is still used for the method. However, a warning is now issued to alert users to the situation. Users can eliminate the warning by setting an explicit transaction attribute for the affected method(s) in the <code>ejb-jar.xml</code> deployment descriptor. Thus, users may now see warnings about missing transaction attribute settings when they run the EJB compiler and during EJB deployment.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR112225	<p>For BLOBs, in the generated code, WebLogic Server calls <code>ObjectOutputStream.writeObject</code> and <code>ObjectInputStream.readObject</code> to serialize/deserialize the object before writing/reading it to the database. These calls add extra header information. The <code>writeObject</code> method writes the class of the object, the signature of the class, the values of the non-transient and non-static fields of the class, and all of its supertypes are written. (This is the reason for the extra header seen in the database.) These calls do not cause a problem when customers are using only WebLogic Server to set and get BLOBs, because it uses <code>readObject</code> to convert the bytes into the appropriate object, which needs that extra header information. However, if the BLOB has been inserted directly into the database by some other vendor or programmer using:</p> <pre>OutputStream os = ((weblogic.jdbc.common.OracleBlob) lob).getBinaryOutputStream(); os.write(this.tiffImage); // byte[] tiffImage</pre> <p>then problems may occur because WebLogic Server uses <code>readObject</code> and the header information is missing. For the data inserted using WebLogic Server, the other programs that would read the bytes directly get the extra header information and fail.</p> <p><code>&lt;serialize-byte-array-to-oracle-blob&gt;</code> has been added to control the persistence behavior. This element is used to specify whether a <code>cmp-field</code> of type <code>byte[]</code> mapped to an <code>OracleBlob</code> should be serialized. The tag has been added to a new compatibility stanza in the <code>weblogic-cmp-rdbms</code> descriptor.</p> <p>Note that, in versions prior to SP2, the default behavior was to serialize a <code>cmp-field</code> of type <code>byte[]</code> mapped to an <code>OracleBlob</code>. Now, the <code>byte[]</code> is written directly to the <code>OutputStream</code> obtained from the BLOB. To revert to the old behavior, set the value of this tag to <code>true</code>.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR112615	<p>Use of the XA driver meant there was no guarantee that WebLogic Server obtained the same connection from the DataSource even when in the same transaction. The Insert was made using one connection and then queried with SELECT SCOPE_IDENTITY using another connection. The SCOPE_IDENTITY produced incorrect results in this case.</p> <p>The auto-key-gen feature for SQLServer2000 required SELECT SCOPE_IDENTITY() to get the value of the auto-gen key.</p> <p>The SCOPE_IDENTITY function can now used in the same scope (that is, in the same stored procedure, function, or batch). The code generation was changed to perform a batched query when the database is SQLServer or SQLServer2000 and AutoKeyGen is turned on. The GenKeyQuery will be fired immediately after the INSERT as follows:</p> <pre data-bbox="292 743 1116 907">__WL_query_array[0] = "INSERT INTO Employees (LastName, FirstName, Title, TitleOfCourtesy, BirthDate, HireDate, Address, City, Region, PostalCode, Country, HomePhone, Extension, Photo, Notes, ReportsTo, PhotoPath) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?) SELECT SCOPE_IDENTITY() ";</pre> <p>and the cmp-field will be assigned the auto generated key value.</p> <p>This change has no impact or side-effects other than resolving the problem of an incorrect primary key being returned when using SQLServer2000 with AutoKeyGen with a XA driver.</p>	8.1 SP2
CR112661	<p>The code to expand the size of the thread pool used for message-driven bean polling was failing due to a change in the "Kernel" APIs.</p> <p>The code for the MDB container was fixed to properly use the kernel APIs. Customers will now be able to deploy more than one MDB that receives messages from a foreign JMS provider with XA and have each MDB successfully receive messages.</p>	8.1 SP2
CR112703	<p>The JMSConnectionPoller did not know the identity of the user that it needed to pass to the WebLogic Security Framework to get the remote username and password. This remote username and password is required for establishing a JMS connection with the remote JMS Server (MQ, WebLogic, and so on).</p> <p>This fix resolves the problem, but customers need to configure a security identity for an MDB. For instructions on how to do this, see <a href="#">"Configuring a Security Identity for Message-Driven Beans"</a> in <i>Programming WebLogic Enterprise JavaBeans</i>.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR113172	<p>When the AutoKey was specified in the <code>cmp-rdbms.xml</code> file as follows:</p> <pre data-bbox="229 461 1016 591">&lt;automatic-key-generation&gt;   &lt;generator-type&gt;ORACLE&lt;/generator-type&gt;   &lt;generator-name&gt;scott.oracle_sequence&lt;/generator-name&gt;   &lt;key-cache-size&gt;10&lt;/key-cache-size&gt; &lt;/automatic-key-generation&gt;</pre> <p>(that is, the generator name is <code>schemaName.sequenceName</code>) WebLogic Server was not taking the schema name into account while performing the query to select the increment value.</p> <p>The schema name provided in the <code>&lt;generator-name&gt;</code> is now taken into account, and the queries are generated as follows:</p> <p>If the users specify the <code>&lt;generator-name&gt;</code> as <code>schemaName.sequenceName</code> in the <code>weblogic-cmp-rdbms-jar.xml</code>, WebLogic Server generates the following query to read the <code>INCREMENT_BY</code>:</p> <pre data-bbox="229 878 973 982">query = "SELECT INCREMENT_BY "+ "FROM ALL_SEQUENCES "+ "WHERE SEQUENCE_NAME = UPPER('"+sequenceName+"') " + " AND SEQUENCE_OWNER = UPPER('"+schemaName+"') ";</pre> <p>If users specify the <code>&lt;generator-name&gt;</code> without the schema, the query will be:</p> <pre data-bbox="229 1043 915 1121">query = "SELECT INCREMENT_BY "+ "FROM USER_SEQUENCES "+ "WHERE SEQUENCE_NAME = UPPER('"+sequenceName+"') ";</pre> <p>In the second query, WebLogic Server makes use of the <code>USER_SEQUENCES</code> view instead of <code>ALL_SEQUENCES</code>. This takes care of the case where two or more users have declared a sequence with the same name. A query to the <code>USER_SEQUENCES</code> view restricts the <code>ResultSet</code> to the user view with a connection to Oracle.</p>	8.1 SP2
CR116696	<p>The <code>setBytes()</code> method on the <code>PreparedStatement</code> had issues when the <code>byte[]</code> data was greater than 4K.</p> <p>This problem was resolved by replacing the <code>setBinaryStream()</code> method with the <code>setBytes()</code> method. As a result, the <code>cmp11</code> generated code has changed. Customers wanting to take advantage of this fix will have to rerun <code>ejbc</code> on their EJBs to generate the new code. Otherwise, this change will not have any impact.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR120257	<p>The EJB container code was not expecting the "Adaptive Server Enterprise" string as the correct database product name for the Sybase database. It was expecting the "Sybase" string in the product name.</p> <p>Because the WebLogic Sybase JDBC driver sends the database product name as "Adaptive Server Enterprise," this string is now considered in the database type verification code of the cmp descriptors.</p>	8.1 SP2
CR120300	<p>An extra AND was appended for ANSI compliant LEFT OUTER JOIN for the following databases: SqlServer2000, Pointbase and DB2.</p> <p>The code was modified not to append AND while generating ANSI compliant LEFT OUTER JOIN for these databases. No side-effects should result from fixing this problem.</p>	8.1 SP2
CR121143	<p>The EJB was correctly failing to deploy because the database table it depended on did not exist. The EJB was configured so the table would be automatically created during deployment if it did not exist. However, the server was in production mode and the auto table creation feature is only supported when the server is in development mode, so the table was not created by the EJB container. There was a problem in that the EJB container was not communicating this information properly to customers.</p> <p>A message was added, explaining auto table creation settings will be disregarded when the server is in production mode. In addition, each Managed Server now logs when an EJB deployment fails because of a missing database table.</p>	8.1 SP2
CR122511	<p>The <code>wls510-read.xpi</code> was not setting the <code>PersistenceMBean.setFindersLoadBean()</code> when the <code>&lt;finders-call-ejbload&gt;</code> element was encountered in the <code>5.1 weblogic-ebb-jar.xml</code> deployment descriptor.</p> <p>When the <code>&lt;finders-call-ejbload&gt;</code> element is encountered in the <code>5.1 weblogic-ebb-jar.xml</code> deployment descriptor, WebLogic Server now sets it on the <code>PersistenceMBean</code>.</p> <pre data-bbox="292 1308 969 1586"> &lt;processing-action element="finders-call-ejbload" element-context="persistence-descriptor"&gt;   &lt;validation nullable="false"     values="true True false False"/&gt;   &lt;java&gt;     if(bd.isEntity()) {       bd.getPersistence().setFindersLoadBean(         "True".equalsIgnoreCase( @VALUE{ } )); }   &lt;/java&gt; &lt;/processing-action&gt; </pre>	8.1 SP2

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR124954	The documentation for the procedure “Configure a Security Identity for a Message-Driven Bean” was missing some steps. This has been corrected, and the updated documentation can be found at <a href="http://edocs.bea.com/wls/docs81/ejb/message_beans.html#ConfiguringaSecurityIdentityforaMessage-DrivenBean">http://edocs.bea.com/wls/docs81/ejb/message_beans.html#ConfiguringaSecurityIdentityforaMessage-DrivenBean</a> .	8.1 SP2

## Installation

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR105921	The state was not preserved on the "Choose BEA Home Directory" panel. The state of controls on the above referenced panel is now retained. Paging back to the "Choose BEA Home Directory" panel will show expected UI control state based on prior user interaction.	8.1 SP2
CR111201	The internal URL format used for database loading was not normalized for all potential runtime platforms. This problem resulted, for example, in a <code>java.lang.IllegalArgumentException</code> when customers attempted to create an e2e domain from the e2e domain template using the Configuration Wizard on Solaris platforms. The URL format used for database loading has been normalized to enable cross platform compatibility.	8.1 SP2

# J2EE

---

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR110964	<p>The J2EEApplicationContainer notifies the listeners of the application deployment phase. During this time, the thread had the system classloader as the contextclassloader.</p> <p>The listener is now notified of the application deployment event with the application classloader as the thread's contextclassloader.</p>	8.1 SP2
CR111138	<p>J2EEApplicationContainer was not notifying the application listeners regarding the "PRESTOP" event on the applications during the server shutdown.</p> <p>Code was added to fire the notifications. Application listeners now receive the PRESTOP notifications.</p>	8.1 SP2
CR122772	<p>JNDI lookup could not resolve an object because the EJB HomeImpl class was not found. This situation resulted in inconsistent assertions and occurred because the HomeImpl class was searched on a classloader different from the one with which the EJB was originally deployed.</p> <p>The J2EEContainer code was fixed to ensure that the CLNode annotation is unique. One annotation should resolve to one CLNode.</p> <p>If this service pack is used, then it should be used in all the clustered servers. Otherwise there are no side-effects.</p>	8.1 SP2

---

## JCOM

Change Request Number	Description	Release Fixed
CR108495	<p>When using <code>com2wls</code>, the security context was not getting propagated between client invocations.</p> <p>A code fix now ensures that a cache is maintained on the server-side when NTLM authentication is not used. When a user is authenticated as a part of the first request from the JCOM client, the client's <code>ThreadContext</code>, the <code>AuthenticatedSubject</code>, and the <code>LastAccessedTime</code> are stored in the cache. If a client remains idle for more than 5 minutes (configured via the <code>weblogic.jcom.clientTimeout</code> property), WebLogic Server removes that client entry from the cache. The trigger frequency is 1 minute by default (configured via <code>weblogic.jcom.invalidationInterval</code> property). This property controls how frequently WebLogic Server runs the trigger to check the cache and clean up idle clients. Note that the values for these properties must be specified in milliseconds.</p>	8.1 SP2
CR120495	<p>DCOM sockets were timing out based on the <code>DefaultNetworkChannel</code>'s idle connection timeout value. This would result in the DCOM socket being closed.</p> <p>Code was added to ensure that the DCOM connections will not timeout.</p>	8.1 SP2

## JDBC

Change Request Number	Description	Release Fixed
CR093038	<p>WebLogic Server now provides support for Oracle Virtual Private Databases (VPDs). A VPD is an aggregation of server-enforced, application-defined fine-grained access control, combined with a secure application context in the Oracle 9i database server.</p> <p>For more information, see "<a href="#">Programming with Oracle Virtual Private Databases</a>" in <i>Programming WebLogic JDBC</i>.</p>	7.0 SP3 8.1 SP2

Change Request Number	Description	Release Fixed
CR104968	<p>WebLogic jDriver for Oracle/XA did not properly handle the NLS_NUMERIC_CHARACTERS parameter set by an Oracle RDBMS instance. This caused a <code>java.sql.SQLException</code> when retrieving double or float values if a comma (`,`') was used as a decimal separator in the database:</p> <pre>java.sql.SQLException: java.lang.NumberFormatException - '1,2' at weblogic.jdbc.oci.ResultSet.getFloat(ResultSet.java:312) at weblogic.jdbc.jta.ResultSet.getFloat(ResultSet.java:190) at weblogic.jdbc.rmi.internal.ResultSetImpl.getFloat(ResultSetImpl.java:224) at weblogic.jdbc.rmi.internal.ResultSetStraightReader.getFloat(ResultSetStraightReader.java:67) at weblogic.jdbc.rmi.SerialResultSet.getFloat(SerialResultSet.java:219) at examples.servlets.DBAccess.service(DBAccess.java:54) at [...]</pre> <p>The problem did not occur with the non-XA version of WebLogic jDriver for Oracle. This problem was solved with a code fix.</p>	6.1 SP6 8.1 SP2
CR106522	<p>The WebLogic Server jDriver for MS SQLServer can mistakenly accept the <code>jdbc:microsoft:...</code> URL for the Microsoft JDBC driver. If a JVM tries to use both drivers, loading the WebLogic Server driver first prevents the Microsoft driver from being used.</p> <p>This jDriver has been deprecated.</p>	7.0 SP4 8.1 SP2
CR107474	<p><code>OciObjectStream</code> did not nullify a buffer after a <code>close()</code> was performed.</p> <p>This problem was resolved with a code fix.</p>	6.1 SP6 8.1 SP2
CR108051	<p>Using LDAP with SSL caused a <code>NullPointerException</code>.</p> <p>This problem occurred because the <code>MuxableSocketLDAP</code> registered itself directly (instead of the <code>SSLFilter</code> for itself) with the socket muxer. Thus, when SSL was used, it was not accessing the right muxable socket instance.</p> <p>This problem was resolved by a change the <code>MuxableSocketLDAP</code> code, so that it registers the <code>SSLFilter</code> with the socket muxer.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR108436	<p>A transactional (JTS) JDBC connection could falsely return true for the <code>isClosed()</code> method in a remote application.</p> <p>The cause of the problem was an unnecessary and failure-prone delegation of the <code>isClosed()</code> call to the remote connection rather than simply reporting the already-known state of the RMI connection.</p> <p>The RMI connection object was modified to return the known state. User code now works as expected with <code>isClosed()</code>.</p>	8.1 SP2
CR109268	<p>When a JNDI lookup on a JDBC DataSource and then a <code>ds.getConnection</code> was performed in applet code, a Security Exception that indicated there was no privilege to generate code in the applet would be thrown.</p> <p>JDBCWrapperFactory now detects whether the lookup is performed within the applet JVM. If it is, rather than generating a JDBC RMI wrapper class in the applet JVM, a request will be sent to <code>ClasspathServlet</code>. <code>ClasspathServlet</code> will generate the JDBC RMI wrapper classes on the server side, then ship them back to the applet. Applets do not work with JDBC RMI without this fix.</p>	8.1 SP2
CR111136	<p>WebLogic Server's Oracle <code>jDriver</code> now understands the non-standard Oracle <code>CURSOR</code> output parameter type, whose numerical value is -10. Customers wanting to use WebLogic Server's Oracle <code>jDriver</code> to accommodate a WebLogic extension will no longer get exceptions.</p>	8.1 SP2
CR111230	<p>During EJB deployment time, certain validations are done to verify the table, check if connection pools exist or not, and so on. If any of these validations fail, a <code>DeploymentException</code> is thrown. In this case, the <code>DeploymentException</code> was being thrown because the <code>TableVerifier.checkPool()</code> method returned <code>false</code>.</p> <p>This problem occurred because in <code>weblogic.jdbc.common.internal.ConnectionPoolManager.poolExists()</code> method, WebLogic Server only checked whether the pool (used by the EJB) existed in the list of available connection pools, and failed to do the same for the list of <code>MultiPools</code>.</p> <p>For the pool associated with the DataSource for the CMP EJB, WebLogic Server now checks that the pool exists in both the connection pool list and the <code>MultiPool</code> list. As a result, customers can use a DataSource mapped to a multipool inside a CMP EJB.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR120224	<p>Many of the Sybase JDBC driver's metadata methods, including <code>getAutoCommit()</code>, interacted with the DBMS in a way that caused the DBMS to think a transaction had been started. This precluded any further changes to transaction properties. For example, the following valid JDBC code caused an application to receive an exception stating: 'can't call set CHAINED within a transaction':</p> <pre> Connection c = ... // get fresh connection c.setAutoCommit(false); // set connection for future multistatement tx's // Now test connection (like after a tx) Change back to default mode if needed if (c.getAutoCommit()) c.setAutoCommit(true); // fails here </pre> <p>Code that is enacted (only) when the initial call to <code>setAutoCommit(true)</code> fails has been added. In this case, WebLogic Server rolls back the transaction and retries the <code>setAutoCommit()</code>. BEA has worked with Sybase to improve their driver, and as of this writing, the latest EBFs of the Sybase driver should include changes that will also fix this problem.</p>	8.1 SP2
CR121635	<p>When running WebLogic Server with the WebLogic Sybase XA JDBC driver, attempts to create a table after suspending the current transaction and starting a new transaction failed, but no exception was thrown.</p> <p>This problem occurred because of cleanup processing WebLogic Server does inside the JDBC layer. When application code calls <code>commit()</code>, WebLogic Server internally checks the Auto Commit setting for the connection. When running with Sybase XA, WebLogic Server needs to call a <code>rollback()</code> because the <code>getAutoCommit()</code> call starts a local transaction. This <code>rollback()</code> also causes the SQL DDL call to create the table to be rolled back. Thus, the table will appear to not have been created.</p> <p>This problem was resolved by removing the call to <code>rollback()</code> after calls to <code>get/setAutoCommit</code> and <code>get/setTransactionIsolation</code> for the Sybase XA driver. Application code now gets the correct results.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR125320	<p>A server thread dump showed a deadlock with one thread calling <code>JTSCConnection.doClose()</code>, which was blocked in this method.</p> <p>This problem occurred because <code>doClose()</code> had a block synchronized on the <code>JTSCConnection</code> with the purpose of ensuring only one thread went into the block at a time. Unfortunately, any other thread running any other synchronized <code>JTSCConnection</code> method would stop a thread getting this <code>doClose()</code> block.</p> <p>A private locking object for the <code>doClose()</code> block was provided. As a result of this fix, there will be no detectable change in behavior, except for fortunate cases where a deadlock is averted and normal function proceeds.</p>	8.1 SP2
CR125496	<p>Oracle-related JDBC operations hung when the <code>ResourcePool</code> maintenance thread locked the resource pool and attempted to obtain a connection that was being held by other threads.</p> <p>This problem occurred because <code>doClose()</code> had a block synchronized on the <code>JTSCConnection</code> with the purpose of ensuring only one thread went into the block at a time. Unfortunately, any other thread running any other synchronized <code>JTSCConnection</code> method would stop a thread getting this <code>doClose()</code> block.</p> <p>A private locking object for the <code>doClose()</code> block was provided. As a result of this fix, there will be no detectable change in behavior, except for fortunate cases where a deadlock is averted and normal function proceeds.</p>	8.1 SP2
CR126038	<p>Oracle OCI NativeXA does not allow a connection to be used by threads other than the one where it is created. The WebLogic Server connection pooling algorithm assumed that a connection could be used by any thread, as this is the behavior of all other JDBC drivers (including the Oracle Thin Driver). Customers who used Oracle OCI NativeXA with this connection pooling algorithm received an <code>XAException</code>.</p> <p>This problem was resolved by pinning the connection to the thread. As a result of this fix, every thread now has its own connection for every connection pool.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR127969	<p>Applications with stringent RASP and HA requirements require Pools to be capable of gracefully recovering from failure conditions such as failure of the database server, the machine hosting the database server, network connectivity to the machine hosting the database server, and so on. The above requires applications to put upper bounds on:</p> <ol style="list-style-type: none"> <li>1. How long a request to create JDBC connections stays blocked</li> <li>2. How long a statement executes</li> </ol> <p>and require support from the JDBC driver; the BEA WebLogic Type 4 JDBC drivers now provide this support. Item 1 is configurable by simply setting the corresponding driver property in the connection pool definition. Item 2 requires support from WebLogic Server and is exposed via the following pool attributes:</p> <ul style="list-style-type: none"> <li>• <code>testStatementTimeout</code>—the time after which the test statement query (specified by applications using the pool attribute <code>TestTableName</code>) currently being executed will be timed-out. The default value implies that this feature is disabled.</li> <li>• <code>statementTimeout</code>—the time after which a statement currently being executed will be timed-out. The default value implies that this feature is disabled.</li> </ul> <p>For more information about the BEA WebLogic Type 4 JDBC drivers, see <a href="#">WebLogic Type 4 JDBC Drivers</a>. For more information about the <code>testStatementTimeout</code> and <code>statementTimeout</code> attributes, see “<a href="#">Timeouts for JDBC Connection Creation and Statement Execution</a>” on page 1-29.</p>	8.1 SP2

## JMS

Change Request Number	Description	Release Fixed
CR103213	<p>When the server failed to send JMS messages, there was no error message on the client. The transaction manager had declared the messages unhealthy, and JMS was rolling back the messages when it failed to enlist resources.</p> <p>The failure to enlist the transaction is now communicated back to client, which will be able to report that the commit failed.</p>	7.0 SP3 8.1 SP2

Resolved Problems for Service Pack 2

Change Request Number	Description	Release Fixed
CR103596	<p>Under a rare situation on a multiprocessor computer, a <code>NullPointerException</code> could occur when a message was acknowledged twice.</p> <p>The code was changed to protect against a <code>NullPointerException</code>. Thus, the <code>NullPointerException</code> will no longer occur in this situation.</p>	7.0 SP4 8.1 SP2
CR104538	<p>A JMS deadlock was fixed by changing the <code>BESession.java</code> <code>close(long lastSequenceNumber)</code> method. The deadlock could be viewed in the partial thread dump:</p> <pre>ExecuteThread: '9' for queue: 'queue_1'" daemon prio=5 tid=0x2757938 nid=0x5e waiting for monitor entry [0xce97f000..0xce97fc68] at weblogic.jms.backend.BETopic._addMessageToConsumers(BETopic java:590) at weblogic.jms.backend.BETopic.addMessageToConsumers(BETopic.j ava:296) at weblogic.jms.backend.BEMessageWriteListener.storeIOComplete( BEMessageWriteListener.java:85) at weblogic.jms.store.StoreRequest.execute(StoreRequest.java:34 2) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:139 ) at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:120) "ExecuteThread: '14' for queue: 'queue_2'" daemon prio=5 tid=0x57b200 nid=0x54 waiting for monitor entry [0xd1b7f000..0xd1b7fc68] at weblogic.jms.backend.BESession.stop(BESession.java:386) at weblogic.jms.backend.BESession.close(BESession.java:434) at weblogic.jms.backend.BESession.close(BESession.java:1083) at weblogic.jms.backend.BESession.invoke(BESession.java:1024) at weblogic.jms.dispatcher.Request.wrappedFiniteStateMachine(Re quest.java:585) ....</pre>	6.1 SP6 8.1 SP2

Change Request Number	Description	Release Fixed
CR105337	<p>Under load conditions, a test JMS application was encountering the following exception:</p> <pre> java.lang.OutOfMemoryError weblogic.jms.common.TransactionRolledBackException: at weblogic.jms.backend.BEConsumer.expireTimeout(BEConsumer.java:1620) at weblogic.jms.backend.BEXATranEntryBlockingConsumer.startRollback(BEXATranEntryBlockin gConsumer.java:72) at weblogic.jms.backend.BEXAResource.rollback(BEXAResource.java:1205) at weblogic.transaction.internal.ServerResourceInfo.rollback(ServerResourceInfo.java:1400) at weblogic.transaction.internal.ServerResourceInfo.rollback(ServerResourceInfo.java:664) at weblogic.transaction.internal.ServerSCInfo.startRollback(ServerSCInfo.java:365) at weblogic.transaction.internal.ServerTransactionImpl.localRollback(ServerTransactionImpl.java: 1521) at weblogic.transaction.internal.ServerTransactionImpl.globalRollback(ServerTransactionImpl.jav a:2142) at weblogic.transaction.internal.TransactionImpl\$1.execute(TransactionImpl.java:1656) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:211) at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:198) weblogic.jms.common.JMSEException: Only one thread may use a JMS Session at a time. at weblogic.jms.frontend.FESession.rollbackAfterRecover(FESession.java:1773) at weblogic.jms.frontend.FESession.recover(FESession.java:1373) at weblogic.jms.frontend.FESession.invoke(FESession.java:2253) at weblogic.jms.dispatcher.Request.wrappedFiniteStateMachine(Request.java:609) at weblogic.jms.dispatcher.DispatcherImpl.dispatchSyncNoTran(DispatcherImpl.java:339) at weblogic.jms.client.JMSSession.recoverGuts(JMSSession.java:868) at weblogic.jms.client.JMSSession.rollback(JMSSession.java:632) at com.ncr.crm.framework.jms.DefaultJmsQueueReceiver.rollback(DefaultJmsQueueReceiver.java: 184) at com.ncr.crm.framework.jms.DemoMessageConsumer.run(JmsMemoryLeakDemo.java:163) at java.lang.Thread.run(Thread.java:479) ----- Linked Exception ----- weblogic.jms.common.JMSEException: Only one thread may use a JMS Session at a time. at weblogic.jms.frontend.FESession.recover(FESession.java:1219) at weblogic.jms.frontend.FESession.invoke(FESession.java:2253) at weblogic.jms.dispatcher.Request.wrappedFiniteStateMachine(Request.java:609) at weblogic.jms.dispatcher.DispatcherImpl.dispatchSyncNoTran(DispatcherImpl.java:339) at weblogic.jms.client.JMSSession.recoverGuts(JMSSession.java:868) at weblogic.jms.client.JMSSession.rollback(JMSSession.java:632) at com.ncr.crm.framework.jms.DefaultJmsQueueReceiver.rollback(DefaultJmsQueueReceiver.java: 184) at...</pre> <p>Research revealed that the WebLogic JMS garbage collection functionality was slow to reclaim memory when dealing with long linked lists; the code was changed to break up the lists. This resolved the problem.</p>	7.0 SP4 8.1 SP2

Change Request Number	Description	Release Fixed
CR106538	<p>When a new Topic Member X was added to a Distributed Topic, consumers on previous Members Y and Z would not get messages published to X. When Member X was on a Migratable Target and added to the Distributed Topic while Y and Z were already booted. Rebooting Y and Z should not be required.</p> <p>This problem was resolved with a code fix to register dynamically added members. Messages published to X are now available on Y and Z.</p>	8.1 SP2
CR106789	<p>A memory leak occurred because unused JMS objects remained open.</p> <p>The problem was resolved by the following fixes:</p> <ul style="list-style-type: none"> <li>• Normalized the destination for all the Producers created on the same destination.</li> <li>• Removed the listener from the correct <code>DispatcherWrapperState</code> when substituting the <code>DispatcherWrapperState</code> for <code>FEProducer</code></li> <li>• Removed the listener from the correct <code>DispatcherWrapperState</code> when closing the <code>FEProducer</code> unconditionally</li> <li>• Removed the leak for Temporary Destinations on <code>FrontEnd</code> when the <code>BackEnd</code> goes down.</li> </ul>	7.0 SP3 8.1 SP2
CR108665	<p>The JMS server deleted an expired message from the persistent store when a queue browser still had the message in its reference list, resulting in a paging IO exception.</p> <p>A code change improved the handling of paging and resolved the problem.</p>	7.0 SP4 8.1 SP2
CR109599	<p>Due to limitations in the Microsoft-provided SQL Server driver, the JMS JDBC store did not work with this driver.</p> <p>The JMS JDBC store software was changed to code around the driver limitation. Thus, the JMS JDBC store can now use the SQL Server driver provided by Microsoft.</p>	7.0 SP4 8.1 SP2

Change Request Number	Description	Release Fixed
CR111172	<p>Foreign JMS Service Provider names were required to be unique. Otherwise, WebLogic Server threw a PROTO error.</p> <p>WebLogic Server has changed the name of the XAResource that is registered by the JMS wrapper code to include the domain name and server name. The JMS TxResource name for <code>foreignServiceProvider</code> has been made unique by adding <code>domain+servername</code> to the <code>resourcename</code>.</p> <p>Thus, the naming convention used to register a foreign JMS provider with an EJB or a servlet has been changed. When using a foreign provider (such as MQSeries) with an EJB or a servlet using the "resource-reference" feature, you must gracefully shut down your WebLogic Server instance to ensure that there are no outstanding two-phase commit transactions before installing this SP. Otherwise, "in-doubt" transactions on the foreign JMS server may not be properly resolved after the server is restarted with the SP the first time. This limitation does not apply to message-driven beans. Also, this precaution is only necessary the first time the server is started after installing this SP, not every time the server is started.</p>	8.1 SP2
CR111538	<p>If <code>res-auth</code> was set to <code>Application</code>, <code>JMSConnectionHelper</code> was not allowing a null value for the username and password.</p> <p>The <code>weblogic.deployment.jms.JMSConnectionHelper</code> constructor has been modified to allow null for the username and password in cases where <code>res-auth</code> is set to <code>Application</code>. Thus, a <code>SecurityException</code> will no longer be thrown for a null username and password if <code>res-auth</code> set to <code>Application</code>.</p>	8.1 SP2
CR112431	<p>The Bytes Maximum and Messages Maximum fields on the Thresholds and Quotas tab for JMS destinations only accepted 32-bit integers, although the help text specified that 64-bit integers were accepted.</p> <p>The validation was corrected to allow for a 64-bit integer.</p>	8.1 SP2
CR121011 CR126883	<p>A <code>NullPointerException</code> resulted on the restart of a JMS Server when a message in the JMS Store had a <code>ReplyTo</code> that was not in the same cluster to which the store belonged and was a <code>Distributed Destination</code>.</p> <p>This problem occurred because WebLogic Server attempted to locate the <code>configMBeanName</code> for the DD member instance that was written to the store, but could not.</p> <p>The code has been modified so that if WebLogic Server cannot locate the <code>configMBeanName</code>, then it will use the instance name and create a <code>DestinationImpl</code> (non-DD queue) for the <code>replyTo</code> field. In other words, WebLogic Server will downgrade the <code>replyTo</code> queue from a <code>DistributedDestination</code> to a normal destination for this scenario. Clients using this <code>replyTo</code> will not get load balancing.</p>	7.0 SP5 8.1 SP2

Change Request Number	Description	Release Fixed
CR122749	<p>Sorting on a <code>DestinationKey JMSCorrelationID</code> could have resulted in a <code>NullPointerException</code> if not all of the messages in the queue had a <code>JMSCorrelationID</code> set. The same failure could have occurred for <code>DestinationKey JMSType</code>.</p> <p>Code was modified to check for a null <code>JMSCorrelationID</code> and null <code>JMSTypes</code> before calling <code>compareTo</code>. As a result, sorting will work correctly, without throwing a <code>NullPointerException</code>.</p>	<p>8.1 SP2</p> <p>6.1 SP6</p>
CR129067	<p>The JMS Server failed to create tables in SQL Server 2K when configured with case sensitive sort order.</p> <p>This problem occurred because JMS was changing the name of tables during creation to all uppercase letters.</p> <p>During creation of JMS tables, table names are now kept as it is, without making all of them upper case. Thus, JMS will now work for all types of SQL Server 2K installations.</p>	8.1 SP2

## JNDI

Change Request Number	Description	Release Fixed
CR103148	<p>A startup class failed to access the local instance <code>MBeanHome</code> with <code>LOCAL_JNDI_NAME</code> (<code>weblogic.management.home.localhome</code>) with <code>LoadBeforeAppDeployments</code> set to true. It failed with the following error message:</p> <pre>Unable to resolve 'weblogic.management.home.localhome' Resolved: 'weblogic.management' Unresolved: 'home'</pre> <p>Moving the bindings of the other JNDI names into the <code>AdminService</code> so that they are available to startup classes fixed the problem.</p>	<p>7.0 SP3</p> <p>8.1 SP2</p>

Change Request Number	Description	Release Fixed
CR120522	<p>When a context is created a <code>TransactionHelperImpl</code> object is pushed onto the stack. When the context is closed the pushed object is popped. While creating a subcontext, the <code>TransactionHelperImpl</code> object was not being pushed onto the stack but when the subcontext was closed, it tried to pop the object resulting in an <code>EmptyStackException</code>.</p> <p>The code was modified to push the <code>TransactionHelperImpl</code> object while creating a subcontext. An <code>EmptyStackException</code> will no longer be seen if subcontexts are created and closed.</p>	8.1 SP2
CR122820	<p>Binding a remote object from a Managed Server to the Administration Server using a startup class and then looking it up resulted in a <code>java.io.StreamCorruptedException</code> on the Managed Server.</p> <p>This problem occurred because the stub of the remote object was being passed over the wire, instead of a dynamically-generated stub from a <code>StubInfo</code> object.</p> <p>The problem was solved by modifying the <code>weblogic.rmi.utils.io.RemoteObjectReplacer.replaceObject</code> method. Now, before writing the stub on the wire, it is replaced with a <code>StubInfo</code> object. As a result of this change, a client that binds a remote object to the JNDI tree of the server will be able to look up that object and invoke methods on it.</p>	8.1 SP2

## JTA

Change Request Number	Description	Release Fixed
CR091974	<p>If XA recovery did not return an XID, WebLogic Server could throw a <code>NullPointerException</code>:</p> <pre>java.lang.NullPointerException at weblogic.transaction.internal.ServerResourceInfo.rollback(Se rverResourceInfo.java:721) at weblogic.transaction.internal.ServerSCInfo.rollback(ServerSC Info.java:318) at weblogic.transaction.internal.ResourceDescriptor.rollbackXid s(ResourceDescriptor.java:1160) at weblogic.transaction.internal.ResourceDescriptor.recover(Res ourceDescriptor.java:1060) at weblogic.transaction.internal.ResourceDescriptor.access\$9(Re sourceDescriptor.java:1029) at weblogic.transaction.internal.ResourceDescriptor\$1.execute(R esourceDescriptor.java:770) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:139 ) at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:120)</pre> <p>The code was modified to check for the presence of an XID and prevent the exception from occurring.</p>	6.1 SP6 8.1 SP2
CR092301 CR107866	<p>A <code>NullPointerException</code> exception could occur during transaction coordination for some configurations. This problem was fixed by a code change.</p>	7.0 SP3 8.1 SP2

Change Request Number	Description	Release Fixed
CR108952	<p>The Sybase JConnect XA driver started local transactions when <code>set/getAutoCommit</code> or <code>set/getTransactionIsolation</code> methods were called. This resulted in errors like the following:</p> <pre>XAER_RMERR : A resource manager error has occurred in the transaction branch javax.transaction.xa.XAException at com.sybase.jdbc2.jdbc.SybXAResource.sendRPC(SybXAResource.java:711) at com.sybase.jdbc2.jdbc.SybXAResource.sendRPC(SybXAResource.java:602) at com.sybase.jdbc2.jdbc.SybXAResource.start(SybXAResource.java:312) at weblogic.jdbc.jta.VendorXAResource.start(VendorXAResource.java:41) at weblogic.jdbc.jta.DataSource.start(DataSource.java:569) at weblogic.transaction.internal.ServerResourceInfo.start(ServerResourceInfo.java:1165) at weblogic.transaction.internal.ServerResourceInfo.xaStart(ServerResourceInfo.java:1108) at weblogic.transaction.internal.ServerResourceInfo.enlist(ServerResourceInfo.java:287) at weblogic.transaction.internal.ServerTransactionImpl.enlistResource(ServerTransactionImpl.java:391) at weblogic.jdbc.jta.DataSource.enlist(DataSource.java:1143) at weblogic.jdbc.jta.DataSource.refreshXAConnAndEnlist(DataSource.java:1098) at weblogic.jdbc.jta.Connection.getXAConn(Connection.java:145) at weblogic.jdbc.jta.Connection.prepareStatement(Connection.java:211)</pre> <p>WebLogic Server now calls <code>connection.rollback()</code> after any invocation of the above methods. Thus, these errors will no longer be seen.</p>	8.1 SP2
CR111719	<p>The attribute <code>KeepXAConnTillTxComplete</code>, which is required for XA connection pools, now only performs a check if the XA isolation level has changed.</p>	7.0 SP4 8.1 SP2

Change Request Number	Description	Release Fixed
CR116697	<p>The following <code>SQLException</code> occurred when customers attempted to migrate their system from WebLogic Server 6.1:</p> <pre>java.sql.SQLException: start() failed on resource 'weblogic.jdbc.wrapper.JTSXAResourceImpl': XAER_RMFAIL : Resource manager is unavailable</pre> <p>This problem occurred because the transaction blocked this <code>XAResource</code> for more than 2 minutes, so the JTA health monitoring system decided to disable its connection pool completely because the <code>XAResource</code> was supposed to be unavailable.</p> <p>This problem was resolved by importing the correct class to allow for the disabling of health monitoring.</p>	8.1 SP2

## Node Manager

Change Request Number	Description	Release Fixed
CR092678	<p>Many parts of WebLogic Server encrypt credentials using a salt. When attempting to reuse the same file with a different salt, a "bad padding exception" (because of a salt mismatch) occurred. This exception was thrown when WebLogic Server was not able to decrypt the password written in a previous incarnation.</p> <p>WebLogic Server now catches this exception and prints a message stating the problem and the corrective action one should take.</p>	8.1 SP2
CR104285	<p>Node Manager's shared object code could cause a segment violation if certain code paths were taken while starting a server instance. The same code paths also failed when using IBM's zLinux JDK. These problems were solved with a code fix to Node Manager.</p>	6.1 SP6 8.1 SP2

Change Request Number	Description	Release Fixed
CR111339	<p>Attempts to start multiple WebLogic Server instances using Node Manager on slower hardware produced the following error:</p> <pre>&lt;Jun 25, 2003 5:16:15 PM CDT&gt; &lt;Error&gt; &lt;NodeManager@192.168.190.4:5555&gt; &lt;__COMMAND_EXCEPTION__Request: failed to execute command 'online' on server 'ifgw1catapp4 - reason: '[JavaProcessControlOnline: Could not get valid pid for WebLogic process.]'&gt;</pre> <p>This problem occurred because Node Manager looked for the PID file before there had been a chance to write to that file.</p> <p>A new property allows users to specify how many times Node Manager will retry reading the PID from the file. Every retry will have a 2 second wait. The property introduced is <code>PIDFileReadRetryCount</code>, which can be specified as a system property or in the <code>nodemanager.properties</code> file. When the value of <code>PIDFileReadRetryCount</code> is greater than zero, Node Manager attempts <code>PIDFileReadRetryCount</code> number of times before giving up on reading the PID from the file. The default value of <code>PIDFileReadRetryCount</code> is zero, so there is no change in default behavior.</p>	8.1 SP2
CR111639	<p>If debug was turned on, the password for starting the Managed Server was being printed in plain text in the Node Manager logs.</p> <p>The password is no longer printed in the logs even if debug is turned on.</p>	8.1 SP2
CR125829	<p>Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03_42.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03_42.00.jsp</a>.</p>	8.1 SP2

## Plug-Ins

Change Request Number	Description	Release Fixed
CR087204	<p>[Apache] In WebLogic Server 6.1 SP03, <code>PathTrim</code> and <code>PathPrepend</code> added an unexpected <code>"</code> at the end of URL. After <code>PathTrim</code> was applied to the original URL and it results to <code>'</code>, an extra <code>'</code> was appended to the <code>PathPrepend</code> variable.</p> <p>This problem was solved with a code fix.</p>	6.1 SP6 8.1 SP2

## Resolved Problems for Service Pack 2

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR086224	<p>(NSAPI) Plug-in did not read the <code>SESSIONID</code> from post data.</p> <p>Analysis revealed a problem in <code>getPreferredServersFromCookie()</code>. When <code>session.getId()</code> was kept as post data, <code>getId()</code> included an extra string: <code> time</code>.</p> <p>The problem was solved by a code fix.</p>	<p>6.1 SP5</p> <p>8.1 SP2</p>
CR091910	<p>(Apache) the Apache plug-in did not read <code>PathPrepend</code> when using <code>&lt;IfModule mod_weblogic.c&gt;</code>. The problem occurred with plug-ins for Apache 1.3.x and Apache 2.0.43.</p> <p>The problem was solved with a code fix.</p>	<p>6.1 SP6</p> <p>8.1 SP2</p>
CR106764	<p>(NSAPI) A thread in the plug-in could obtain a critical lock for a long duration (5 minutes by default, or configured using <code>WLIOTimeoutSecs</code>), blocking all other threads and making WebLogic Server appear to hang. This problem was fixed with a code change to the plug-in.</p>	<p>6.1 SP6</p> <p>8.1 SP2</p>
CR107254	<p>(Apache) Because Hewlett-Packard ceased to support the HP Apache-based Web Server version 1.3.x, WebLogic Server removed the Apache 1.3 plug-in for HP.</p>	<p>7.0 SP3</p> <p>8.1 SP2</p>
CR108092	<p>(ISAPI) The ISAPI plug-in logged an unhandled exception error in the Windows event log when it encountered a modified cookie. The event text began with the line:</p> <p>The HTTP Server encountered an unhandled exception while processing the ISAPI Application.</p> <p>This problem was solved with a code fix to the ISAPI plug-in.</p>	<p>6.1 SP6</p> <p>8.1 SP2</p>
CR108747	<p>(NSAPI) The plug-in that shipped with WebLogic Server used the <code>WL-Proxy-Client-IP</code> header to send client IP addresses. This caused problems in heterogeneous environments, because earlier servers expected the plug-in to use the <code>X-Forwarded-For</code> and <code>Proxy-Client-IP</code> headers to transmit client IP addresses.</p> <p>The code was fixed to include both the <code>X-Forwarded-For</code> and <code>Proxy-Client-IP</code> headers, as well as <code>WL-Proxy-Client-IP</code>, so that heterogeneous environments can retrieve client addresses without modification.</p>	<p>6.1 SP6</p> <p>8.1 SP2</p>
CR109755	<p>(Apache) The plug-in ignored configuration parameters that contained regular expressions other than wildcard characters (<code>* / ?</code>). This could cause 404 errors to occur when using parameters such as:</p> <p><code>LocationMatch "/weblogic/(abc def)/ghi"</code></p> <p>This problem was solved with a code fix.</p>	<p>6.1 SP6</p> <p>8.1 SP2</p>

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR120806	(Apache) Not passing back the status from the Java command made it impossible to tell if the command executed successfully or not.  The line: <code>if errorlevel 1 exit /b 1</code> was added after the two Java commands in the <code>ant.bat</code> file. Therefore, the status of the Java commands are now passed back to the shell if they fail.	8.1 SP2
CR121341	Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03_39.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03_39.00.jsp</a> .	6.1 SP6 7.0 SP5 8.1 SP2
CR123149	(All) As of WebLogic Server 6.1 SP6, 7.0 SP5, and 8.1 SP2, the WebLogic plug-ins are now certified to proxy to any version of WebLogic, including 5.1.	8.1 SP2

## RMI/RMI-IIOP

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR124806	Multiple problems were found in iiop tunneling - <ul style="list-style-type: none"> <li>- Assertion violation when registering for pending responses due to HttpURLConnection retrying some http requests when tunneling.</li> <li>- Thin client attempting a direct connection to the cluster nodes instead of going through the plug-in or load balancer</li> <li>- Method descriptor corruption due to thread safety problem.</li> </ul> The above mentioned problems were fixed by adding code to <ul style="list-style-type: none"> <li>- Detect duplicate http requests made by HttpURLConnection and discard duplicate pending responses.</li> <li>- Ensure that all requests made by iiop thin client during tunneling will go through the proxy.</li> <li>- Ensure that iiop initialization is synchronized.</li> </ul>	

Change Request Number	Description	Release Fixed
CR081853	Fixed some complicated problems with RMI-IIOP and the marshalling of complex objects, notably in environments using a mixture of JDKs.	6.1 SP5 8.1 SP2
CR106281	<p>Replicated session beans under heavy load resulted in increasing heap usage and <code>OutOfMemoryErrors</code>.</p> <p>Analysis revealed that <code>examples.ejb20.basic.statefulSession.TraderBean_5ysgq2_EOImpl</code> objects were not being garbage-collected even when garbage collection was forced. For clustered stateful session beans, strong references were maintained to the <code>EOImpl</code> object primary in the <code>weblogic.rmi.cluster.PrimarySecondaryRemoteObject</code>. As a <code>remove</code> is never called on the bean, the reference to the <code>EOImpl</code> was never removed from the <code>eoMap</code>.</p> <p>A code fix was implemented to unexport the <code>EO</code> when the passivated bean has been deleted, after <code>session-timeout-seconds</code>.</p>	6.1 SP6 8.1 SP2
CR107547	<p>When using JMS and the thin-client, connection timeouts were turned off on the server side. This left the possibility of clients not being detected as dead when a network was partitioned.</p> <p>Timeouts are now switched back on, and the server pings the client just before closing the connection. If the ping succeeds, then the connection is not closed. As a result of this change, partitions are correctly detected when using the thin-client.</p>	8.1 SP2
CR108317	Some accessors did not map properly in the ejbc-generated IDL (Interface-Definition-Language). The problem has been resolved.	7.0 SP3 8.1 SP2
CR112648	<p>Tuxedo clients would crash when trying to decode a <code>valuetype</code> contained within an <code>any</code>. This problem occurred because the <code>TypeCode</code> in the <code>any</code> was set incorrectly.</p> <p>The <code>TypeCode</code> of the <code>any</code> is now set to that of the contained <code>valuetype</code>. As a result of this change, Tuxedo no longer crashes.</p>	8.1 SP2
CR112791	<p>Servers send component keys with host port information. Clients use that host port information to make further remote requests. In WebLogic Server 8.1, clients did not have enough information on what scheme the connection was created with, and therefore used default scheme of HTTP.</p> <p>The code has been modified so that the <code>TunneledSSLOrBSocketFactory</code> will be created initially with the HTTPS scheme. This fix forces clients to always use an HTTPS scheme when in the context of <code>TunneledSSLOrBSocketFactory</code> for all succeeding requests.</p>	8.1 SP2

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR116690	<p>The IIOP encoding routines had an element which involved linear search. This became very expensive for large data objects.</p> <p>The linear search problem has been removed. Marshaling large objects is now fast over RMI-IIOP.</p>	8.1 SP2
CR116692	<p>Linear search was being used for decoding indirections. This became very expensive for large objects.</p> <p>The linear search problem has been removed. Marshaling large objects is now fast over RMI-IIOP.</p>	8.1 SP2
CR120180	<p>When using the HTTPS protocol with thin clients and network channels (IIOPS tunneling), the server did not honor the secure protocol when sending default network channel information back to the client. As a result, the client received the response and started talking to the server directly using the information in the response, hence creating another handshake.</p> <p>The Connection code was modified so that it can identify whether the protocol being used is secure or not. As a result of this change, the server will correctly identify that the established connection is secure and use the correct network channel. Clients will make succeeding requests properly.</p>	8.1 SP2
CR120593	<p>When a WebLogic Server instance went down, clients did not realize it and attempted to connect to the server by sending Tunnel Recv requests. This caused additional traffic to the server when it came up. At the same time, the execution of existing JMS sessions caused the server to create a new connection with the same ID as the previous one. As a result, the server received the requests on the same connection (recv followed by send) and tried to register a pending response when it already had one, producing an <code>AssertionError</code>.</p> <p>A code fix now causes clients to realize that the server has died and to close the existing connection. Therefore, there is no way the client can use the existing JMS session to connect to server when the server comes back up again. The client will now get <code>IllegalStateExceptions</code> if it reuses the same JMS session after a server restart.</p>	8.1 SP2
CR120898	<p>Customers using the thin-client and tunneling experienced poor performance for large objects. This was caused by the JDK ORB splitting messages into 1k packets.</p> <p>The ORB has been configured to not use fragmentation when doing tunneling. As a result of this change, customers should experience up to 30% performance improvement with tunneling in the thin-client.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR121005	<p>Customers would like to register <code>valuetype</code> factories for base types. WebLogic Server could allow this by publishing repository ID lists for <code>valuetypes</code>, but there were some bugs in the handling of these. In addition, the IBM ORB sends repository ID lists.</p> <p>Bugs in the IIOP decoder were fixed and a new <code>IIOPBean</code> property, <code>UseFullRepositoryIdList</code>, was introduced. When customers set the <code>UseFullRepositoryIdList</code> property to <code>true</code> in <code>config.xml</code>, the WebLogic Server-IIOP runtime will send full <code>RepositoryId</code> lists for <code>valuetypes</code>. This enables C++ ORB users to register base types for decoding. However, <code>RepositoryId</code> lists are incorrectly handled by JDK 1.4.x and earlier versions of Tuxedo 8.1, so this property should only be set in exceptional circumstances.</p>	8.1 SP2
CR121079	<p>When making an initial request, the 8.1 WebLogic Server-IIOP runtime would not select the appropriate codeset for wide string transmission.</p> <p>The appropriate codeset for wide string transmission is now selected for all requests. Interoperability with WebLogic Server 8.1 will now work correctly.</p>	8.1 SP2
CR121309	<p>A regression was introduced in WebLogic Server 8.1 that prevented WebLogic Server from selecting an appropriate codeset for wide string transmission. It would always default to UCS-2.</p> <p>The appropriate codeset selection is now made. Interoperability with foreign ORBs such as WebSphere should now work correctly.</p>	8.1 SP2
CR121501	<p>The Java to IDL specification requires that interfaces of the form:</p> <pre data-bbox="229 1121 848 1229">public interface FooInterface {     public void mymethod() throws IOException; }</pre> <p>be treated as abstract interfaces just as</p> <pre data-bbox="229 1286 901 1394">public interface FooInterface {     public void mymethod() throws RemoteException; }</pre> <p>already is.</p> <p>The abstract interface detection routine was changed to correctly detect interfaces of this type. Interfaces of this type can now be used with the thin-client and other RMI-IIOP ORBs.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR121897	<p>The IIOP implementation in WebLogic Server incorrectly serialized <code>String.class</code> (the class itself, not an instance of the class).</p> <p>The serialization of <code>String.class</code> has been fixed so it can now be used in RMI-IIOP requests.</p>	8.1 SP2
CR122478	<p>The <code>ORB.string_to_object()</code> method was handled incorrectly for foreign initial references.</p> <p>The problem was fixed by ensuring that the initial reference is encoded in the generated IOR. As a result of this change, the <code>ORB.string_to_object()</code> method now behaves correctly.</p>	8.1 SP2

## Security

Change Request Number	Description	Release Fixed
CR085094	<p>After creating a domain with one Administration Server and one Managed Server, enabling the <code>-Djava.security.manager</code> for the two servers returned a <code>java.io.Filepermissions</code> error reported in the AdminServer log for <code>&lt;Error&gt; &lt;EmbeddedLDAP&gt; &lt;000000&gt; &lt;Error parsing Entry #19: access denied (java.io.FilePermission.\myserver\ldap\ldapfiles\EmbeddedLDAP.data read)&gt;</code>.</p> <p>This problem was resolved by updates to the <code>weblogic.policy</code> file, to which was added information about granting permissions to user applications, and examples of grant statements. The <code>weblogic.policy</code> works for Weblogic Server Examples, Pet Store, and Workshop. It needs to be modified to be used for user configurations; instructions are in the comments at the top of the file.</p>	7.0 SP3 8.1 SP2
CR095836	<p>The default value for backup minutes is now correctly calculated and reported under domain <code>&gt; Domain Wide Security Settings -&gt; Embedded LDAP</code> in the Administration Console.</p>	7.0 SP3 8.1 SP2

Change Request Number	Description	Release Fixed
CR096934	<p>Construction of an <code>AuthenticationProvider</code> was not recognizing errors, because WebLogic MBeanMaker was not seeing exceptions thrown by the <code>AuthenticationMBI</code> class constructor.</p> <p>The problem was resolved by a code change ensuring the propagation of errors.</p>	7.0 SP3 8.1 SP2
CR099759	<p>The <code>weblogic.security.acl.SSLUserInfo.getCertificates()</code> method was not added to the <code>SSLUserInfo</code> class for WebLogic Server 8.1 or 8.1 SP1, even though it was added for 7.0 SP2.</p> <p>The <code>getCertificates()</code> method has been added to <code>SSLUserInfo</code> for 8.1 SP2.</p>	8.1 SP2
CR101652	<p>In previous releases of WebLogic Server, it was not possible to make a secure connection to a third-party XML server because the XML server did not support the SSL protocol.</p> <p>The following workaround is provided:</p> <ol style="list-style-type: none"> <li>1. Create a WebLogic client (a JavaServer page (JSP) deployed in WebLogic Server).</li> <li>2. Create an Java authenticator class according to the Java specification. This authenticator should obtain a surname and password for the used to connect to the XML server.</li> <li>3. Specify Net Permission for the Java authenticator class in the <code>weblogic.policy</code> file. The Net Permission is granted to all users of that Java VM. Note this presents a possible security weakness and should therefore be used carefully.</li> <li>4. Use <code>java.net.HttpURLConnection</code> to establish a connection to the XML server.</li> <li>5. Specify <code>-DUseSunHttpHandler</code> as a command-line argument when starting WebLogic Server.</li> </ol>	7.0 SP3 8.1 SP2
CR103405	<p>SSL initialization failed with a Not listening for SSL, <code>java.io.IOException: Inconsistent security configuration, null error</code>. This occurred when the parsing of the server certificate resulted in a null modulus &amp; exponent.</p> <p><code>RSAPublicKey.toString()</code> did not check for a null value, and was therefore trying to call the <code>toString()</code> method of its null member variables. This caused the method to fail with a <code>NullPointerException</code>.</p> <p>A modification to the <code>toString()</code> method to check for null values fixed this problem.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR105234	<p>The WebLogic Server Administration Console hung while attempting to list the users and groups in an Active Directory.</p> <p>This problem occurred because the connection was hanging with LDAP results, causing a problem in iterating those results.</p> <p>The code was modified to release the connection with search results and was upgraded to new Netscape open source API. As a result of this change, the listing of users and groups in the Administration Console no longer hangs.</p>	8.1 SP2
CR106027	<p>Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03-28.01.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03-28.01.jsp</a>.</p>	7.0 SP3 8.1 SP2
CR106357	<p>When multiple Authorization providers were installed, you could not define security policies on Web applications.</p> <p>This problem has been fixed.</p>	7.0 SP3 8.1 SP2
CR106404	<p>If you created a user that contained special characters such as "&amp;", "(, ", or "* ", then the user would be created in the Embedded LDAP server, but an error message would be displayed. For example, if you created a user with name <code>myuser&amp; (</code>, then the following error message would be displayed:</p> <pre data-bbox="292 1017 1096 1069">weblogic.management.utils.NotFoundException: User or Group myuser&amp; (</pre> <p>The user <code>myuser&amp; (</code> would be added to the Embedded LDAP server, but from then on, no users would be displayed in the console.</p> <p>This problem was resolved by ensuring that special characters are escaped when retrieving the users from the Embedded LDAP server. This allows the user names with special characters to be created and displayed correctly. User names with special characters can now be successfully created and displayed by the console.</p>	8.1 SP2
CR108886	<p>Clients sending a request to a WebLogic Server instance were not able to parse the returned SOAP response.</p> <p>This problem occurred because WebLogic Server was using a static variable to hold the prefix that the runtime uses for the SOAP envelope.</p> <p>The output stream now picks up the prefix for the SOAP namespace from the stream. WebLogic Server looks for the namespace currently in use by the SOAP stack. There is also a <code>SecureSoapOutputStream</code> constructor that allows customers to specify a preferred SOAP envelope namespace.</p>	8.1 SP2

## Resolved Problems for Service Pack 2

Change Request Number	Description	Release Fixed
CR112886	<p>The identity assertion naming convention (<code>WL-PROXY-CLIENT-*</code>) for headers conflicted with <code>WL-PROXY-CLIENT-KEYSIZE</code>, <code>WL-PROXY-SECRET-KEYSIZE</code> and <code>WL-PROXY-CLIENT-IP</code>.</p> <p>Filtering these headers as identity assertion headers avoided a Client-Cert identity assertion error.</p>	<p>7.0 SP4</p> <p>8.1 SP2</p>
CR113459	<p>The Node Manager properties file is always created in the <code>&lt;saved logs dir&gt;/NodeManagerInternal</code> directory. Some customers periodically archive and delete the contents of <code>NodeManagerInternal</code>. This removes the Node Manager properties file, so that the certificate password stored in the Node Manager properties file cannot be decrypted, and Node Manager will not work correctly.</p> <p>The problem was solved by a code change to create the Node Manager properties file in the directory specified by <code>user.dir</code>, if the file is not found in <code>NodeManagerInternal</code>, or in the <code>user.dir</code> directory.</p>	<p>6.1 SP6</p> <p>8.1 SP2</p>
CR120265	<p>When using <code>webserviceclient.jar</code> and <code>ssl.jar</code> in a standalone client application, the entire SOAP envelope was being written to <code>stdout</code> for every call. Attempts to disable verbose mode on the command-line and programmatically did not resolve the problem.</p> <p>This problem occurred because <code>URLConnection</code> was printing out the unnecessary log. This logging has been deleted, so <code>URLConnection</code> no longer shows post data on the <code>system.out</code>.</p>	8.1 SP2
CR120460	<p>The ServerIron monitoring product sends a request to the SSL port during a server health check. During this health check, the <code>SSLIOConect</code> and related SSL objects were not released, causing an out-of-memory error.</p> <p>This known problem is fixed in this release.</p>	<p>7.0 SP2</p> <p>8.1 SP2</p>
CR123801	<p>The MedRec sample application has been enhanced to support identity assertion in the <code>LoginModule</code>.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR125446	<p data-bbox="292 418 1099 473">Performance regressed from WebLogic Server 7.0 when using SSL Web Services because socket pooling was disabled.</p> <p data-bbox="292 487 1130 718">In WebLogic Server 7.0, SSLSockets were pooled in the Web Service runtime, in the thread safe socket pool that is a singleton shared across all invocations in the VM. In cases where SSL Client Authentication is used, this may lead to a problem: If there are multiple clients invoking the service within the application with different identities, the second client will possibly get the SSL Socket from the first client—this connection on this socket carries the first client identity. This is potentially a severe problem when using the client in a server setting—a transaction may occur while authenticated under the wrong identity, and this may lead to unprivileged users executing privileged actions.</p> <p data-bbox="292 732 1130 817">The shared socket pool has been replaced with a single socket that is shared only for a single client in the HTTPS binding. The socket sharing is disabled by default, however it can be enabled using either the system property or the new API provided.</p> <p data-bbox="292 831 1130 944">The default operational characteristics out of the box remain the same as WebLogic Server 8.1, with socket sharing disabled. Turning this feature on allows multiple serial invocations to share the same socket, but has the following properties that users must understand and accept:</p> <ul data-bbox="292 958 1130 1442" style="list-style-type: none"> <li data-bbox="292 958 1130 1043">• The Web Service generated client stubs are thread safe. When this socket sharing feature is enabled the client is no longer thread safe, therefore users must manage access of the client by multiple threads.</li> <li data-bbox="292 1057 1130 1142">• The client has security properties (such as client certs and sharing sockets) that require careful thought. Additionally, users will also be tying up JVM resources, and are responsible for closing the socket.</li> <li data-bbox="292 1156 1130 1442">• Socket sharing is "false" by default. Socket sharing is enabled using the following system property (boolean): <code>https.sharedsocket</code>. Socket sharing can also be enabled using the following API (referenced in <code>weblogic.webservice.binding.https.HttpsBindingInfo</code>) <code>setSocketSharing(boolean value)</code>. Socket sharing carries a timeout value based in seconds. This value can be set using the following system property: <code>https.sharedsocket.timeout</code>. Socket sharing timeout is defaulted to 15 seconds. Socket sharing timeout can also be set by using the following API: <code>setSharedSocketTimeout(long value)</code> where <code>value</code> is in seconds. The shared socket can also be closed using <code>void closeSharedSocket()</code>.</li> </ul>	8.1 SP2
CR125457	<p data-bbox="292 1468 749 1494">Please review the security advisory information at</p> <p data-bbox="292 1498 948 1524"><a href="http://dev2dev/resourcelibrary/advisoriesnotifications/BEA03_43.00.jsp">http://dev2dev/resourcelibrary/advisoriesnotifications/BEA03_43.00.jsp</a>.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR125829	Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03_42.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03_42.00.jsp</a> .	8.1 SP2
CR190838	WebLogic Server Version 8.1 uses the SUN JDK 1.4 which does not support RSA keys larger than 2048 bits. Therefore, certificates with 4096 bit keys can not be used.	

## Servlets & JSPs

Change Request Number	Description	Release Fixed
CR084785	The <code>GenericProxyServlet</code> can now handle wrapped responses.	6.1 SP4 8.1 SP2
CR088525	In previous WebLogic Server 8.1 Service Packs, JSP parameters with a value of <code>"/"/</code> were interpreted by the <code>getParameter()</code> as having the value <code>"/</code> . The problem was resolved with a code change.	7.0 SP3 8.1 SP2
CR090465	When hosting a Web application with user-defined error pages, WebLogic Server returned the following response for 400 error and closed the connection immediately, with no <code>Connection: close</code> header: <pre>HTTP/1.1 400 Bad Request Date: Thu, 01 Jan 1970 00:00:00 GMT Content-Length: 463 Content-Type: text/html Last-Modified: Thu, 07 Nov 2002 21:56:52 GMT &lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN"&gt; &lt;html&gt;</pre> The problem was solved with a code fix.	6.1 SP5 8.1 SP2
CR091472	A JSP compilation exception propagated to <code>weblogic.Deployer</code> by the server did not contain the exact error message and just had a stack trace. The exact compilation error, including the JSP error, has now been propagated to <code>weblogic.Deployer</code> . Thus, the exception has more details and context information.	8.1 SP2

Change Request Number	Description	Release Fixed
CR091831	<p>The WebLogic Server implementation of <code>URLConnection</code> did not check whether keep-alive connection had been timed out on the server side when using POST method, resulting in the error: Connection aborted by peer: socket write error on flush.</p> <p>Turning off keep-alive on the server made the problem disappear. Increasing Keep-Alive periods still showed the problem with different length of sleep.</p> <p>Checks were added to ensure that the <code>HttpClient</code> is non-null before updating the timestamp.</p>	7.0 SP3 8.1 SP2
CR092039	<p>The <code>JspParser</code> threw an <code>UnsupportedEncodingException</code> for the extra quoted value in charset with JSP tag:</p> <pre data-bbox="292 739 1096 762">&lt;%@ page contentType="text/html; charset=\"Shift_JIS\"" %&gt;</pre> <p>resulted in this error:</p> <pre data-bbox="292 817 1123 921">java.lang.RuntimeException: Unknown/unsupported charset: "Shift_JIS" - java.io.UnsupportedEncodingException: Charset: '"Shift_JIS"' not recognized, and there is no alias for it in the WebServerMBean</pre> <p>WebLogic Server did not correctly support the HTTP specification.</p> <p>The problem was resolved by adding support for quoted strings and comments to Content-Type header charset attribute value.</p>	6.1 SP5 8.1 SP2
CR092625	<p>WebLogic Server threw a <code>NullPointerException</code> when you enabled HTTP logging on a Managed Server that was booted with HTTP logging disabled and had no existing log file. On each HTTP access to the Managed Server, the following exception was thrown:</p> <pre data-bbox="292 1161 1123 1494">java.lang.NullPointerException at weblogic.servlet.logging.LogManagerHttp.log(LogManagerHttp.j ava:292) at weblogic.servlet.internal.HttpServer.log(HttpServer.java:865 ) at weblogic.servlet.internal.ServletResponseImpl.send(ServletRe sponseImpl.java:1044) at weblogic.servlet.internal.ServletRequestImpl.execute(Servlet RequestImpl.java:2265) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:139 ) at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:120)</pre> <p>The problem was solved with a code fix.</p>	6.1 SP6 8.1 SP2

Change Request Number	Description	Release Fixed
CR093014	<p>Java comments that start in one scriptlet and end in the next one cause an error:</p> <pre data-bbox="229 458 1056 631">&lt;22.07.2002 14:12:24 CEST&gt; &lt;Error&gt; &lt;HTTP&gt; &lt;[WebAppServletContext(7422744,DefaultWebApp,/DefaultWebApp) ] Servlet failed with Exception weblogic.servlet.jsp.JspException: (line 12): scriptlet close brace '}' unbalanced at line 12 which breaks scope '_base_service_scope_' at ...</pre> <p>The problem was solved by back-porting grammar from WebLogic Server 7.0 that caused the ScriptletScopeLexer to skip an entire Java comment.</p>	6.1 SP5 8.1 SP2
CR093520	<p>When you precompiled JSPs on a machine in one time zone, and then deployed those same JSPs on a server in a different time zone, WebLogic Server sometimes re-compiled the JSPs. This occurred because WebLogic Server checked JSPs by comparing the local timestamp of the JSPs (as embedded by the JAR utility) against the timestamps in the generated class files.</p> <p>The problem was resolved by storing the time zone at compile time and using that time zone at deployment time to determine whether recompilation is necessary.</p>	6.1 SP6 8.1 SP2
CR093625	<p>WebLogic Server misdirected output for JSP includes in JSP pages. The problem was solved with changes to the servlet engine to unify handling of includes, forwards, wrapped-responses, and JSP BodyTags.</p>	6.1 SP5 8.1 SP2
CR093649	<p>The ejb2jsp utility was generating incorrect tag classes for methods that contained parameter types other than String.</p> <p>The problem was fixed by changing the tag library generator to use <code>getAttribute</code> rather than <code>getAttributeString</code>.</p>	7.0 SP3 8.1 SP2
CR094190	<p>According to the JVM specification, the size limit for a method size is 64K. In WebLogic Server, a customer has JSPs with many body tags for which the generated Java code exceeded the 64K limit for the <code>__jspService</code> method.</p> <p>The problem was solved by a code change. Now, when an empty <code>BodyTag</code> of the form <code>&lt;my:tag ... /&gt;</code> is encountered, the generated code that would normally set up scope for the body is replaced by a single call to a static method in <code>StandardTagLib.java</code> that tricks the <code>BodyTag</code> into thinking it is being invoked as usual from the JSP source.</p>	6.1 SP5 8.1 SP2

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR094252	<p>Displaying error/warning messages with staged copies of WAR files made it difficult to decipher the actual WAR/Web application being deployed.</p> <p>Now, the application name is always appended to the module name in all descriptor validation messages. As a result of this change, error messages look different from previous versions. No other side-effects are envisioned.</p>	8.1 SP2
CR094997	<p>When two interacting Web applications use different session cookie names, the session ID was lost.</p> <p>Analysis revealed that session attributes were not updated correctly when a Web application specifies <code>cookieName</code>, a resource from another Web application is included, and a <code>RemoteUser</code> exists in the request prior to the inclusion of the external resource.</p> <p>The problem was solved by a code fix.</p>	6.1 SP5 8.1 SP2
CR096041	<p>WebLogic Server no longer uses redirection (HTTP status code 302) when returning a Welcome file.</p> <p>WebLogic Server versions prior to 8.1 SP2 and 7.0SP4 were returning a 302 (moved temporary) status code and the location of the Welcome file.</p> <p>WebLogic Server versions 8.1 SP2, 7.0 SP4 and later return a 200 (OK) status code displaying the content of the Welcome file.</p> <p>WebLogic Server performance is improved as a result of this change, however the relative URL in the Welcome file may point to a different location which would result in a 404-Not Found error.</p>	7.0 SP4 8.1 SP2

Change Request Number	Description	Release Fixed
CR096399	<p>JSP compilation errors were handled improperly. The filter communicated with <code>OutputStream</code> instead of with <code>HttpServletRequestWrapper</code> <code>getWriter</code>, with this result:</p> <pre>java.lang.IllegalStateException: strict servlet API: cannot call getWriter() after getOutputStream() at weblogic.servlet.internal.ServletResponseImpl.getWriter(ServletResponseImpl.java:171) at weblogic.servlet.internal.ServletRequestImpl.reportJSPFailure(ServletRequestImpl.java:227) at weblogic.servlet.internal.ServletRequestImpl.reportJSPCompilationFailure(ServletRequestImpl.java:239) at weblogic.servlet.jsp.JspStub.reportCompilationFailure(JspStub.java:523) at weblogic.servlet.jsp.JspStub.compilePage(JspStub.java:430) at weblogic.servlet.jsp.JspStub.prepareServlet(JspStub.java:210) at weblogic.servlet.jsp.JspStub.prepareServlet(JspStub.java:164) at weblogic.servlet.internal.ServletStubImpl.getServlet(ServletStubImpl.java:517) at weblogic.servlet.internal.ServletStubImpl.invokeServlet(ServletStubImpl.java:351) at weblogic.servlet.internal.ServletStubImpl.invokeServlet(ServletStubImpl.java:445) at weblogic.servlet.internal.TailFilter.doFilter(TailFilter.java:20) at weblogic.servlet.internal.FilterChainImpl.doFilter(FilterChainImpl.java:27) at WebLogicServerBugFilter.doFilter(WebLogicServerBugFilter.java:27) at weblogic.servlet.internal.FilterChainImpl.doFilter(FilterChainImpl.java:27) at weblogic.servlet.internal.WebAppServletContext\$ServletInvocationAction.run(WebAppServletContext.java:5418) at weblogic.security.service.SecurityServiceManager.runAs(SecurityServiceManager.java:744) at weblogic.servlet.internal.WebAppServletContext.invokeServlet(WebAppServletContext.java:3086) at weblogic.servlet.internal.ServletRequestImpl.execute(ServletRequestImpl.java:2544) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:153) at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:134)...</pre> <p>WebLogic ServerBugFilter.doFilter after</p> <p>The problem was fixed by the implementation of a wrapper-aware <code>RequestCallback</code>.</p>	7.0 SP3 8.1 SP2
CR096459	<p>When invoking a flush in a Servlet or JSP in HTTP 1.0, WebLogic Server sometimes failed to close the socket, causing the client to wait for a period of time until the socket timed out.</p> <p>The problem was solved with a code fix.</p>	6.1 SP6 8.1 SP2

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR098388	<p>Right after server startup, some server requests were rejected because the MBean did not update server state before the listening thread started receiving requests.</p> <p>This problem has been resolved by using Boolean flags inside the Web container to update the server state for the listening thread.</p>	<p>7.P SP3</p> <p>8.1 SP2</p>
CR098518	<p>The following EJB/Web application setup was causing a class cast exception:</p> <pre>Servlet Filter1 --&gt; WebApp1 -- forward to WebApp2 ctx --&gt; Servlet Filter2(WebApp2) --&gt; Ejb Lookup (fails)</pre> <p>This problem has been fixed by locating the EJB lookup in the Servlet instead of a Servlet Filter when doing a forward between two Web applications.</p>	<p>7.0 SP3</p> <p>8.1 SP2</p>
CR101838	<p>A change in the behavior of the <code>CGIServlet</code> that could affect the execution of CGI scripts having no filename extension. For example, if a <code>web.xml</code> file defined the CGI directory and Servlet mapping in the following way:</p> <pre>&lt;param-name&gt;cgiDir&lt;/param-name&gt; &lt;param-value&gt;/home/user/cgi-bin&lt;/param-value&gt; &lt;/init-param&gt; &lt;/servlet&gt; &lt;servlet-mapping&gt; &lt;servlet-name&gt;CGIServlet&lt;/servlet-name&gt; &lt;url-pattern&gt;/cgi-bin/*&lt;/url-pattern&gt; &lt;/servlet-mapping&gt;</pre> <p>and stored a script named <code>myscript</code> in the <code>/home/user/cgi-bin</code> directory, a URL such as <code>http://localhost/mywebapp/cgi-bin/myscript</code> would map to <code>/home/user/cgi-bin</code> without specifying the script name. (This problem did not occur with scripts having an extension, such as <code>myscript.ksh</code>).</p> <p>The code was fixed to make the behavior match earlier releases of <code>CGIServlet</code>. Using the above example, the code fix ensures that the URL <code>http://localhost/mywebapp/cgi-bin/myscript</code> maps to <code>/home/user/cgi-bin/myscript</code>.</p>	<p>6.1 SP6</p> <p>8.1 SP2</p>
CR102036	<p>The first access to deleted JSP pages was causing a <code>JspFileNotFoundException</code> instead of 404 File Not Found. Further access to this page correctly returned 404.</p> <p>The problem is fixed so that deleted JSP pages always return 404.</p>	<p>7.0 SP3</p> <p>8.1 SP2</p>

Change Request Number	Description	Release Fixed
CR102628	<p>The following JSP code:</p> <pre data-bbox="229 458 1059 708"> &lt;jsp:plugin type="applet" code="examples.applets.PhoneBook1.class" codebase="/bea_WebLogic Server_internal/classes/DefaultWebApp@DefaultWebApp/" height="800" width="500" type="applet" jreversion="1.3.1_06" nspluginurl="http://java.sun.com/products/plugin/1.1.3/plugin-install.html"; iepluginurl="http://java.sun.com/products/plugin/1.1.3/jinstall-113-win32.cab#Version=1,1,3,0" &gt; </pre> <p>generated this HTML:</p> <pre data-bbox="229 765 1059 927"> &lt;embed type="application/x-java-applet;version=1.3.1_06" pluginspage="http://java.sun.com/products/plugin/1.1.3/plugin-install.html"; height="800" width="500" code="examples.applets.PhoneBook1.class" codebase="/bea_WebLogicServer_internal/classes/DefaultWebApp@DefaultWebApp/"&gt; </pre> <p>The generated HTML code failed on Netscape: Netscape attempted to download the plug-in from WebLogic Server instead from the SUN site.</p> <p>When the plug-ins page line was moved before the codebase line, the code worked correctly on Netscape.</p> <p>A code fix to order the applet attributes properly resolved the problem.</p>	<p>6.1 SP6 8.1 SP2</p>

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR102675	<p>If a taglib is imported once via the taglib directive in a page that includes a second page, and that second page also imports that same taglib, then invalid XML results in the following browser output:</p> <pre>Error 500--Internal Server Error From RFC 2068 Hypertext Transfer Protocol -- HTTP/1.1: 10.5.1 500 Internal Server Error The server encountered an unexpected condition which prevented it from fulfilling the request.</pre> <p>And the following log file output:</p> <pre>&lt;[ServletContext(id=287644,name=jspVal.war,context-path=/jsp Val)] Servlet failed withIOException java.io.IOException: javax.servlet.jsp.JspException: The taglib validator rejected the page: "Exception TestValidator XML parsing: org.xml.sax.SAXParseException: Attribute "xmlns:valtest" was already specified for element "jsp:root". -- stack trace written to stderr., "...</pre> <p>Because of a code change to <code>weblogic.servlet.jsp.jsp2xml.Jsp2XmlOutputter</code>, duplicate tag prefixes are ignored in the generation of the XML views of the JSPs.</p>	7.0 SP3 8.1 SP2
CR102769	<p>Forwarding a request from a servlet or JSP to another JSP page did not log <code>queryString</code> in the <code>access.log</code>. Additionally, <code>access.log</code> logged the forwarded request, but not the original request.</p> <p>A code change to <code>RequestDispatcherImpl</code> fixed the problem.</p>	7.0 SP3 8.1 SP2
CR103256	<p>A code change resulted in performance improvement in JDBC regarding bubble caches in <code>JDBCSessionContext</code> and <code>JDBCSessionData</code>.</p>	6.1 SP6 8.1 SP2
CR103304	<p><code>jsp_precompile</code> now works for all kinds of JSPs, including those that extend custom classes.</p>	7.0 SP4 8.1 SP2

Change Request Number	Description	Release Fixed
CR103214	<p>When you set the <code>compilerclass</code> JSP parameter, WebLogic Server logged a <code>compilerclass=null</code> message in the startup log, as in:</p> <pre>JSPServlet with initArgs '[JspConfig: verbose=true,packagePrefix=jsp_servlet,-compiler= C:\61sp4\jdk131\bin/javac,compileFlags=,workingDir=C:\61sp4 wlserver6.1\config\examples\applications\examplesWebApp\WEB- INF\_tmp_war_examples_examplesWebApp, pageCheckSeconds=1,superclass=weblogic.servlet.jsp.JspBase, keepgenerated=false,precompileContinue=false, compilerSupportsEncoding=true,encoding=null,defaultfilename index.jsp,compilerclass= null,noTryBlocks=false]'</pre> <p>The problem occurred because the setting of <code>compilerclass</code> was not used during startup. The correct setting was used for compiling JSPs. The code was fixed to obtain the correct parameter value during server startup.</p>	6.1 SP6 8.1 SP2
CR103339	<p>Servlet output capitalized the <code>Transfer-Encode</code> value, <code>Chunked</code>, instead of using lowercase <code>chunked</code> as required by the HTTP/1.1 specification. The code was fixed to ensure that Servlets use the <code>Transfer-Encode</code> type of "chunked" in lowercase.</p>	6.1 SP6 8.1 SP2
CR103719	<p>When users log in to a protected page through HTTPS, the client receives two cookies back: <code>JSESSION</code> and <code>_wl_authcookie_</code>. When the server sends the second request (HTTPS) to an unprotected page without <code>_wl_authcookie_</code>, a 401 error was returned.</p> <p>An issue where access was denied to an unprotected (that is, no security restrictions) HTTPS page when the browser client did not send back the <code>_wl_authcookie_</code> has been fixed. Unprotected pages should no longer require the cookie.</p>	8.1 SP2
CR103861	<p>Applications deployed with relative paths caused <code>UnsafeFilenameExceptions</code> because WebLogic Server did not resolve the parent directory.</p> <p>The code was modified so that when WebLogic Server attempts to obtain a safe file name, it checks the parent directory first, and resolves it if needed.</p>	8.1 SP2
CR106072	<p>The <code>pageCheckSeconds</code> attribute, which sets the interval, in seconds, at which WebLogic Server checks to see if JSP files have changed and need recompiling, did not work the first time a JSP was modified.</p> <p>This problem occurred because WebLogic Server did not set a <code>lastStaleCheck</code> time the first time a JSP was invoked.</p> <p>The code was modified so that if <code>lastStaleCheck</code> is not set yet, it is set to the current time. This prevents the JSP from being recompiled unnecessarily.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR106226	<p>Code that used a response wrapper and called <code>getOutputStream</code> on the original response passed into a JSP resulted in <code>NestedBodyResponse</code> always throwing an <code>IllegalStateException</code>.</p> <p>Mixed use of <code>getWriter()</code> and <code>getOutputStream()</code> from <code>NestedBodyResponse</code> are now allowed, so the exception will only be thrown when appropriate.</p>	8.1 SP2
CR106577	<p>In prior releases, WebLogic Server served requests containing extra spaces in the URI by trimming off the ending spaces. As of WebLogic Server 8.1 SP2, however, extra spaces in a URI will now result in a 404 (resource not found).</p> <p>For example:</p> <pre>http://server:port/mywebapp/foo%20%20</pre> <p>used to resolve to the resource <code>foo</code> in the Web application <code>mywebapp</code>. This URI will now result in a 404.</p> <p>No backward compatibility issues are expected.</p>	8.1 SP2
CR106856	<p>When customers using servlet filtering in their application do a forward or include via the request dispatcher, WebLogic Server ran the request through the filter again and caused an infinite loop. According to the 2.3 servlet specification, this is undefined behavior; in the 2.4 servlet specification, the default behavior is not to refilter the request on forward and include when processing via the request dispatcher.</p> <p>A <code>filter-dispatched-requests-enabled</code> configuration parameter has been added to <code>weblogic.xml</code> to enable customer control of servlet filtering.</p>	8.1 SP2
CR107419	<p>JSP code that intended to set double byte characters as a parameter's value—for example:</p> <pre>&lt;jsp:include page="included.jsp"&gt;   &lt;jsp:param name="title"     value="&lt;%= java.net.URLEncoder.encode(\" [double byte       characters]\" )     %&gt;" /&gt; &lt;/jsp:include&gt;</pre> <p>—resulted in specified double byte characters being changed to their URL-encoded code.</p> <p>This problem has been resolved.</p>	6.1 SP6 8.1 SP2

Change Request Number	Description	Release Fixed
CR108046	<p><code>weblogic.net.http.HttpURLConnection</code> did not implement the <code>getHeaderFields()</code> method that exists in JDK 1.4. As a result, <code>weblogic.net.http.HttpURLConnection.getHeaderFields()</code> returned an empty map.</p> <p>The implementation of <code>getHeaderFields()</code> has now been added to <code>weblogic.net.http.HttpURLConnection</code>. As a result of this change, <code>weblogic.net.http.HttpURLConnection.getHeaderFields()</code> now correctly returns a map of header fields.</p>	8.1 SP2
CR108607	<p>WebLogic Server produced errors when applications used XSL with the FOP output method to get an image from an archive file. This problem did not occur for applications deployed in exploded format.</p> <p>This problem was solved with a code fix.</p>	6.1 SP6 8.1 SP2
CR109479	<p>The <code>JnlpDownloadServlet</code> uses the timestamp on a WAR file to determine whether the client's resources are older than what is available on the Web server. To get the timestamp, <code>JnlpDownloadServlet</code> calls the <code>URLConnection.getLastModified()</code> method. However, the <code>getLastModified()</code> method returned 0.</p> <p>The code has been modified so that a <code>ZipURLConnection.getLastModified()</code> method returns the modification time of the zip file, and thus the WAR file.</p>	7.0 SP5 8.1 SP2
CR109586	<p>Load-on-startup Servlets were being constructed (static initializers and constructor) as the kernel user.</p> <p>The static initializers, constructs and the <code>init</code> method will now be invoked as:</p> <ul style="list-style-type: none"> <li>• <code>init-as</code> user, if specified</li> <li>• <code>run-as</code> user, if specified</li> <li>• anonymous user</li> </ul>	8.1 SP2
CR109821	<p>IndexFile resolution, which does expensive IO operations, was happening in the pre-dispatch phase (muxer queue).</p> <p>The code was modified to defer this resolution until after the dispatch phase.</p>	8.1 SP2
CR109958	<p>When processing requests through a proxy servlet, WebLogic Server only honored the <code>SecureProxy</code> setting for incoming requests that used HTTPS. If the incoming request used HTTP, WebLogic Server did not use an HTTPS connection even when <code>SecureProxy</code> was enabled in the proxy servlet. This problem was solved with a code fix.</p>	6.1 SP6 8.1 SP2

Change Request Number	Description	Release Fixed
CR110293	<p>The session count was an approximate number but was not initialized back to 0 when the server was restarted. Also the increment/decrement was not synchronized. As a result, a race condition could have contributed to a wrong count. In addition, for replicated sessions the count was not incremented/decremented for <code>becomePrimary</code>, <code>becomeSecondary</code>, and <code>becomeUnregistered</code> callbacks.</p> <p>The code was modified to obtain the count from the map directly for memory and replicated sessions. For file-based sessions, the count is obtained from the file system and for JDBC, it is obtained via a database query. As a result of this fix, counts will not be broken anymore. Note, however, that the runtime MBean returns now two more counts:</p> <ol style="list-style-type: none"> <li>1. Total sessions opened by the server so far.</li> <li>2. The maximum open sessions in memory.</li> </ol> <p>These two counts remain unaffected by this change. They will still use the old method of increment and decrement; they are approximate and represent a number since the time the server was restarted. These two counts will be reset to 0 after the server restarts.</p>	8.1 SP2
CR110914	<p>If <code>TrackingEnabled</code> was set to <code>false</code>, WebLogic Server created a new session for each request, but the sessions were not getting invalidated.</p> <p>The code was modified to invalidate a session immediately if <code>TrackingEnabled</code> is set to <code>false</code>. This is the correct behavior.</p>	8.1 SP2
CR111406	<p>If session data was accessed after redeployment—even with the <code>save-sessions-enabled</code> property set to <code>true</code>—a <code>java.lang.NoClassDefFoundError</code> occurred.</p> <p>This problem has been resolved with a code fix.</p>	8.1 SP2
CR111600	<p>The <code>request.isUserInRole</code> method was not querying the WebLogic Security Framework's Role Manager for security roles when no users were logged in.</p> <p>The <code>request.isUserInRole</code> method will now query the Role Manager for security roles even when no users have logged in.</p>	8.1 SP2
CR112413	<p>When the Web application container reloads a modified servlet class dynamically, it calls the <code>destroy()</code> method on the old instance. However, the <code>destroy</code> was not being invoked with the Web application classloader as the thread's <code>ContextClassLoader</code>.</p> <p>The code has been modified so that the <code>destroy()</code> method is always invoked with the Web application classloader as the thread's <code>ContextClassLoader</code>.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR112547	<p>When the number of servers in the cluster was large, it was difficult to configure the same property for all of them.</p> <p>To make this easier, the properties <code>FrontendHost</code>, <code>FrontendHTTPPort</code> and <code>FrontendHTTPSPort</code> in the <code>ClusterMBean</code> have been exposed. The <code>ClusterMBean</code> values will take effect only for the default Web Servers of the servers in the cluster. Virtual host properties need to be set separately. This fix does not result in any change to existing behavior.</p>	8.1 SP2
CR112799	<p>WebLogic Server attempted to write to an output stream even after an <code>IOException</code> occurred. This led to 100% CPU utilization if an unexpected socket disconnection occurred with a Web Application that did not handle <code>IOException</code>.</p> <p><code>org.apache.xml.serialize.XMLSerializer</code> ignores <code>IOException</code> until the end of its process. This caused the problem to occur if an <code>IOException</code> was thrown in the middle of returning XML documents as part of an HTTP response.</p> <p>The code was modified to ensure that writes to an output stream stop after an <code>IOException</code> occurs.</p>	6.1 SP6 8.1 SP2
CR112803	<p>When running <code>appc</code>, an <code>Exception</code> stating that the <code>ejb-ref</code> had no mapping to the JNDI name of the target EJB was incorrectly thrown.</p> <p>This problem occurred because of a logic error in the Web application compliance checker. The logic was corrected to correctly recognize when an <code>ejb-ref</code> declared in <code>web.xml</code> has a corresponding JNDI name mapping in <code>weblogic.xml</code>. As a result, <code>appc</code> will no longer incorrectly throw this exception.</p>	8.1 SP2
CR112992	<p>If XML schema caching was enabled and a JSP was refreshed, it threw an exception starting with the following lines:</p> <pre>weblogic.servlet.jsp.JspException: (line -1): cannot load TLD: weblogic.xml.dom.ChildCountException: missing child tagclass in tag ...</pre> <p>This problem occurred because the caching caused the entity not to be resolved, which made the <code>TLDDescriptor</code> assume that the TLD was 1.1, when in fact the TLD was 1.2.</p> <p>The <code>TLDDescriptor</code> constructor now takes a boolean flag <code>_12</code> and an <code>org.w3c.dom.Element</code>, then calls <code>Document.getDoctype()</code> to inspect the type of the document.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR114936	<p>WebLogic Server drained an inputstream even if a client cancelled a proceeding request. It would therefore consume CPU resources.</p> <p>The code was changed so that connections are thrown away so they do not drain an inputstream. As a result of this change, if an <code>IOException</code> occurs while writing to a client, <code>GenericProxyServer</code> does not recycle this connection and will create a new connection for a subsequent request.</p>	7.0 SP5 8.1 SP2
CR116209	<p>If customers added a Serializable <code>ServletContext</code> attribute, then overwrote it with a non-Serializable value, the original value masked the new value. This problem occurred because the old value was never removed from the attributes Hashtable.</p> <p>The code has been modified to remove the value from an attributes Hashtable, if necessary, when replacing a Serializable value with a non-Serializable one.</p>	8.1 SP2
CR111752	<p>Under certain conditions, WebLogic Server threw a <code>NullPointerException</code> when using <code>CGIServlet</code> with the <code>useByteStream</code> parameter set to true. The problem occurred when using framesets where one frame contained static URL links and another frame used <code>CGIServlet</code>. If a user selected the frame containing static links before the other frame completed downloading a page, an IO exception was caught and presented to the user as:</p> <pre data-bbox="292 996 1089 1052">java.lang.NullPointerException at weblogic.utils.Executable\$Drainer.run(Executable.java:366)</pre> <p>This problem was solved with a code fix.</p>	6.1 SP6 8.1 SP2
CR120630	<p>Weblogic Portal 8.1 calls the <code>RequestDispatcher.include()</code> method many times using an original <code>ServletRequest</code>, which extends <code>ServletRequestWrapper</code>. To handle a <code>ServletRequestWrapper</code> object, WebLogic Server needs to do casting. In the <code>RequestDispatcherImpl.include()</code> method, WebLogic Server attempted this casting operation and caught a <code>ClassCastException</code> if the referenced <code>ServletRequest</code> object was not an instance of the <code>ServletRequestImpl</code> class. This caused performance degradation when using WebLogic Portal 8.1 because the performance cost of the exception generated by an unpermitted cast attempt is expensive.</p> <p>The code has been modified so that the <code>instanceof</code> operator is used instead of the <code>ClassCastException</code> to determine whether or not a specific casting operation is permitted.</p> <p>As a result of this change, the <code>weblogic.servlet.internal.RequestDispatcherImpl.forward()</code> and <code>include()</code> methods use the <code>instanceof</code> operator for flow control.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR121748	<p>Due to a problem with Supplementary Policies support, the temporary directory (which normally contains JSP class files) would be deleted whenever an application was undeployed. This temporary directory was not intended to be deleted unless the application was explicitly removed.</p> <p>A code fix solved the problem.</p>	8.1 SP2
CR125108	<p>In WebLogic Server 6.x, the URL pattern: <code>/foo*</code> matched <code>/foo</code>. This was not legal per the Servlet 2.3 specification.</p> <p>In WebLogic Server 7.x the <code>URLResource</code> was introduced, and WebLogic Server unintentionally stopped supporting this. In addition, due to eccentricities in the Servlet container, the <code>/foo*</code> URL pattern worked if the <code>fullyDelegateAuthorization</code> flag was disabled, but not if it was enabled. Users would view this as a degradation in security when upgrading from WebLogic Server 6.x to 7.x or 8.x.</p> <p>With this fix, only a string beginning with a <code>/</code> character and ending with a <code>/*</code> postfix is used for path mapping. Since customers may rely on the old mapping, WebLogic Server does not deploy Web applications with a <code>&lt;url-pattern&gt;</code> set to <code>/foo*</code> in <code>&lt;security-constraints&gt;</code>, to prevent a security vulnerability.</p> <p>As a result of this fix, a URL pattern like <code>/foo*</code> will be treated as an exact match, and Web applications with a <code>&lt;url-pattern&gt;</code> set to <code>/foo*</code> in <code>&lt;security-constraints&gt;</code> will not be deployed.</p>	8.1 SP2
CR126319	<p>When a server in a cluster is (forcibly) shut down, the server invalidates the primary and secondary sessions, thereby causing session failover to fail.</p> <p>Code to check the cluster size when a session was destroyed upon server shutdown was available. (That is, if the server was not the last server in the cluster, the session was not invalidated.) However, WebLogic Server did not pass the <code>serverShutdown</code> flag correctly so the checking logic was ignored.</p> <p>Code was modified so that the server state is checked, and the checking logic is invoked when destroying a session.</p>	8.1 SP2

# System Administration

---

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR063279	When multiple elements in WebLogic Server's <code>config.xml</code> had the same objectnames, later elements overwrote configurations for preceding elements. This used to happen silently. Now, WebLogic Server throws an error message (ID=150012) to standard output.	7.0 SP3 8.1 SP2

---

Change Request Number	Description	Release Fixed
CR099474	<p>The <code>getAttributes()</code> method in the <code>RequiredModelMBean</code> should return a <code>javax.management.AttributeList</code>, which should contain <code>javax.management.Attribute</code> objects, but in the WebLogic Server JMX implementation the <code>AttributeList</code> returned by <code>RequiredModelMBean</code> contained the value of the MBean attribute directly and not as a <code>javax.management.Attribute</code> object, causing the following exception:</p> <pre> java.lang.ClassCastException: java.lang.String at weblogic.management.internal.RemoteMBeanServerImpl.getAttributes(RemoteMBeanServerImpl.java:210) at weblogic.management.internal.RemoteMBeanServerImpl_WebLogicServerkel.invoke(Unknown Source) at weblogic.rmi.internal.BasicServerRef.invoke(BasicServerRef.java:362) at weblogic.rmi.internal.BasicServerRef\$1.run(BasicServerRef.java:313) at weblogic.security.service.SecurityServiceManager.runAs(SecurityServiceManager.java:821) at weblogic.rmi.internal.BasicServerRef.handleRequest(BasicServerRef.java:308) at weblogic.rmi.internal.BasicExecuteRequest.execute(BasicExecuteRequest.java:30) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:213) at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:189) End server side stack trace at com.adventnet.beasupport.client.JMXConverter.convertInvocationTargetException(JMXConverter.java:744) at com.adventnet.beasupport.client.ProxyMBeanServer.getAttributes(ProxyMBeanServer.java:511) at com.adventnet.beasupport.client.WebLogicClient.getAttributes(WebLogicClient.java:593) at com.adventnet.beasupport.client.WebLogicClient.cmdline(WebLogicClient.java:1279) at com.adventnet.beasupport.client.WebLogicClient.access\$000(WebLogicClient.java:15) at com.adventnet.beasupport.client.WebLogicClient\$2.run(WebLogicClient.java:1293) </pre> <p>The <code>GetAttributes</code> method now returns an <code>Attribute</code> object.</p>	7.0 SP3 8.1 SP2

Change Request Number	Description	Release Fixed
CR100756	<p>Setting a JMSServer's Store property via the <code>weblogic.Admin</code> utility:</p> <pre>java weblogic.Admin -url t3://localhost:7001 -username system -passwd weblogic SET -mbean examples:Name=examplesJMSServer,Type=JMSServer -property Store examples:Name=examplesJMSFileStore,Type=JMSStore</pre> <p>resulted in the following exception:</p> <pre>&lt;Mar 10, 2003 3:32:15 PM EST&gt; &lt;Error&gt; &lt;Management&gt; &lt;141033&gt; &lt;Error importing MBean examples:Name=examplesJMSFileStore,Type=JMSStore to server examplesServer.javax.management.InstanceNotFoundException: examples:Name=examplesJMSFileStore,Type=JMSStore</pre> <p>This problem was partially fixed by checking when setting an attribute that is a <code>WebLogicObjectName</code> on a configuration MBean. The code checks to see if the bean is a valid one and will throw an <code>InvalidAttributeValueException</code>. This is logged on the Administration Server and the JMSServer remains usable but does not have the file store set.</p>	7.0 SP3 8.1 SP2
CR105736	Some load-balancing applications such as Cisco Loadbalancer became unable to use session	7.0 SP3
CR121053	stickiness because of a change to the SessionID format.	8.1 SP2
	<p>A new server startup flag,  <code>-Dweblogic.servlet.useExtendedSessionFormat=true</code>, retains the information that the load-balancing application needs for session stickiness.</p>	
CR105777	Exception propagation was not working properly because <code>RemoteException</code> was not being treated as a valuetype for rapid caching.	7.0 SP3 8.1 SP2
	Caching only real valuetypes fixed the problem.	
CR105831	An exploded Web application was allowed to load classes from the docroot.	7.0 SP3
	This inappropriate classloading behavior has been disallowed.	8.1 SP2
CR106198	On <code>GroupReaderMBean</code> , <code>isMember</code> was failing when member name contained a comma.	7.0 SP3 8.1 SP2
	A code change resolved this problem so that member names containing commas are now accepted.	

## Resolved Problems for Service Pack 2

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR106728	<p>The log handlers assumed that only <code>WLLogRecord</code> instances were being published. In the case of users directly invoking the logger using the Java 1.4 Logging APIs, the handlers threw a <code>ClassCastException</code> because it got a <code>java.util.logging.LogRecord</code> instance instead of a <code>weblogic.logging.WLLogRecord</code>.</p> <p>A <code>java.util.logging.LogRecord</code> is now converted to a <code>WLRecord</code> object before being processed by the handlers. This avoids the <code>ClassCastException</code>. Users can now invoke the JDK Logger directly using the Java 1.4 Logging APIs and messages are logged to the server log file.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR107805	During server startup in a cluster on WebLogic Platform, the following error occurred:	8.1 SP2
CR109620	<p data-bbox="292 461 1130 1459">           &lt;Jun 18, 2003 4:46:53 PM EDT&gt; &lt;Error&gt; &lt;Management&gt; &lt;BEA-141009&gt; &lt;Error occurred in the file distribution servlet while processing request type "wl_ear_resource_request", java.net.SocketException: Software caused connection abort: socket write error. java.net.SocketException: Software caused connection abort: socket write error at java.net.SocketOutputStream.socketWrite0(Native Method) at java.net.SocketOutputStream.socketWrite(SocketOutputStream.java:92) at java.net.SocketOutputStream.write(SocketOutputStream.java:136) at weblogic.servlet.internal.ChunkUtils.writeChunkNoTransfer(ChunkUtils.java:263) at weblogic.servlet.internal.ChunkUtils.writeChunks(ChunkUtils.java:228) at weblogic.servlet.internal.ChunkOutput.flush(ChunkOutput.java:296) at weblogic.servlet.internal.ChunkOutput.checkForFlush(ChunkOutput.java:371) at weblogic.servlet.internal.ChunkOutput.write(ChunkOutput.java:241) at weblogic.servlet.internal.ChunkOutputWrapper.write(ChunkOutputWrapper.java:114) at weblogic.servlet.internal.ServletOutputStreamImpl.write(ServletOutputStreamImpl.java:171) at weblogic.management.servlet.FileDistributionServlet.returnInputStream(FileDistributionServlet.java:568) at weblogic.management.servlet.FileDistributionServlet.doGetEarResourceRequest(FileDistributionServlet.java:787) at weblogic.management.servlet.FileDistributionServlet.doGet(FileDistributionServlet.java:500) at javax.servlet.http.HttpServlet.service(HttpServlet.java:740) at javax.servlet.http.HttpServlet.service(HttpServlet.java:853) at weblogic.servlet.internal.ServletStubImpl\$ServletInvocationAction.run(ServletStubImpl.java:1053) at weblogic.servlet.internal.ServletStubImpl.invokeServlet(ServletStubImpl.java:387) at weblogic.servlet.internal.ServletStubImpl.invokeServlet(ServletStubImpl.java:305) at weblogic.servlet.internal.WebAppServletContext\$ServletInvocationAction.run(WebAppServletContext.java:6310) at weblogic.security.acl.internal.AuthenticatedSubject.doAs(AuthenticatedSubject.java:317) at weblogic.security.service.SecurityManager.runAs(SecurityManager.java:118) at weblogic.servlet.internal.WebAppServletContext.invokeServlet(WebAppServletContext.java:3622) at weblogic.servlet.internal.ServletRequestImpl.execute(ServletRequestImpl.java:2564) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:197) at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:170)         </p> <p data-bbox="292 1472 696 1494">This socket write error has been eliminated.</p>	

Change Request Number	Description	Release Fixed
CR110161	<p>For any <code>weblogic.Admin</code> command that connects to a WebLogic Server instance, you must provide user credentials. You can now use the new <code>STOREUSERCONFIG</code> command to encrypt the user credentials instead of passing credentials directly on the command line or storing unencrypted credentials in scripts. See <a href="#">STOREUSERCONFIG</a> for more information.</p>	8.1 SP2
CR112123	<p>In WebLogic Server 8.1 and below, <code>Weblogic MBeans (SecurityConfigurationMBean)</code> could not have references to <code>Security MBeans (RealmMBean)</code>. As a result, the realm could not be an attribute on the <code>SecurityConfigurationMBean</code>. Instead, it had to be an operation (a finder) as opposed to a getter. This was problematic because only users in the <code>Administrators</code> group have access to operations (except for special cases).</p> <p>The <code>findRealm()</code> and <code>findRealms()</code> methods on the <code>SecurityConfigurationMBean</code> are now special-cased. As a result, these two methods are now publicly accessible.</p>	8.1 SP2
CR112324	<p>Three commands—<code>wlconfig</code>, <code>wldeploy</code>, and <code>wlserver</code>—were modified so they can obtain a username and password from secured locations.</p> <p>For <code>wlconfig</code> and <code>wldeploy</code>, customers can use the User Configuration functionality as described for the <code>weblogic.Admin</code> tool. The command: <code>STOREUSERCONFIG option wlserver</code> uses the <code>boot.properties</code> file to obtain a username and password.</p> <p>As a result, usernames and passwords can be omitted from ant scripts, as long as an administrator has created the user config/key files (for <code>wlconfig/wldeploy</code>), or the server has been booted at least once in the domain (for <code>wlserver</code>). These actions create the files in which the encrypted usernames and passwords are stored. The username and password can still be provided on the ant task command line. In this case, they are used (that is, the user config or <code>boot.properties</code> files are ignored). In short, existing ant task scripts will continue to work as usual, but administrators who want to remove the username/password from the scripts can use the new functionality of these commands.</p>	8.1 SP2
CR113475	<p>A <code>NullPointerException</code> occurred when security MBeans attempted to de-encrypt the encrypted security MBean attributes during initialization.</p> <p>This problem occurred because WebLogic Server MBeans were not yet initialized, and the encryption service relied on getting the salt from these MBeans.</p> <p>The <code>NullPointerException</code> is now caught and the salt is obtained from the domain log file.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR124392	<p>There were multiple issues associated with this CR.</p> <ol style="list-style-type: none"> <li>1. <code>SecurityConfiguration.findDefaultRealm()</code>, <code>findRealms()</code>, and <code>findRealm()</code> always turned to the Administration Server for this information.</li> <li>2. Whenever a <code>CommoProxy</code> object was initialized, it tried to obtain a reference to <code>AdminMBeanHome</code> in its constructor, although this was not required until later for setters and certain operations.</li> <li>3. For read-only operations that required access to Embedded LDAP, WebLogic Server would always attempt to obtain this information from the Administration Server's LDAP. Whenever the Administration Server was down, exceptions were thrown in different layers.</li> <li>4. Even after these problems were fixed, the response was slow when the Administration Server was down (what took about 30 seconds would have taken 3-5 seconds if the Administration Server was running). This was because WebLogic Server attempted to initialize the <code>adminMBeanHome</code> in <code>CommoProxy</code> in its constructor.</li> </ol> <p>These problems were solved as follows:</p> <ul style="list-style-type: none"> <li>• <code>SecurityConfiguration</code> now obtains necessary information from the local MBean Server.</li> <li>• WebLogic Server defers the initialization of <code>AdminServerMBeanHome</code> until it is actually required.</li> <li>• For certain operations, WebLogic Server now always goes to the local MBean Server as opposed to the Administration Server.</li> </ul>	8.1 SP2
CR122568	<p>A startup class failed to access the local instance's MBeanHome. It failed with the following error message when the startup class was deployed with <code>LoadBeforeAppDeployments="true"</code>:</p> <pre>Unable to resolve 'weblogic.management.home.localhome' Resolved: 'weblogic.management' Unresolved: 'home'</pre> <p>This problem occurred because WebLogic Server 8.1 SP1 could not get the local instance's MBeanHome with <code>LOCAL_JNDI_NAME</code> (<code>weblogic.management.home.localhome</code>).</p> <p><code>LOCAL_JNDI_NAME</code> is now available to startup classes.</p>	8.1 SP2

## Tools

## Resolved Problems for Service Pack 2

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR074637	WebLogic Builder did not provide a way to remove a default transaction. Customers can now delete a default transaction by selecting the blank line in the combo box.	8.1 SP2
CR074714	In the WebLogic Builder WebLogic Settings panel, when a RAR's "Capacity increment" was larger than the difference between "Maximum capacity" and "Initial capacity", WebLogic Builder failed to load the RAR, and printed a SAXProcessorException beginning: weblogic.xml.process.: capacity-increment should be less than or equal to (max_capacity-initial_capacity) A code change resolved the problem.	7.0 SP3 8.1 SP2
CR083151	Command-line DDInit now works for EAR files.	7.0 SP3 8.1 SP2
CR093658	Calling <code>getLogFileName</code> on the <code>VirtualHostMBean</code> was returning an incorrect path.	7.0 SP3 8.1 SP2
CR093833	Redeploying to a Managed Server using <code>weblogic.Deployer</code> without specifying the server was deploying the application to the Administration Server. The problem has been resolved by a change to the <code>weblogic.Deployer</code> utility.	7.0 SP3 8.1 SP2
CR098134	<code>weblogic.Admin</code> prompted for a password, and then threw an exception before you could type it. A code change resolved this problem.	7.0 SP3 8.1 SP2
CR099461	WebLogic Builder did not prompt users about what the valid entries might be for CMP fields. Instead, these had to be hand-typed by the user. WebLogic Builder now populates a menu that allows users to select entries for CMP fields.	8.1 SP2
CR100538	WebLogic Builder threw a <code>NullPointerException</code> if users attempted to open a <code>ejb.jar</code> file that did not have an <code>assembly-descriptor</code> stanza in its <code>ejb-jar.xml</code> file. The <code>NullPointerException</code> is now guarded against, and users of WebLogic Builder can open an <code>ejb.jar</code> file where the <code>ejb-jar.xml</code> has no <code>assembly-descriptor</code> stanza.	8.1 SP2

Change Request Number	Description	Release Fixed
CR103188	<p>When WebLogic Builder was loading the Web application deployment descriptors, the <code>resource-ref</code> stanza was being deleted. This problem was caused by a bug in the servlet descriptor processing code.</p> <p>The servlet descriptor processing code has been modified so that it properly reads the <code>resource-ref</code> stanza. Web applications with a <code>resource-ref</code> stanza in their deployment descriptors are now loaded properly.</p>	8.1 SP2
CR107916	<p>Deleting and adding back a CMP field shuffled the DD CMP field and column mappings. This problem occurred because when editing an existing CMP Field, the field-name is <code>readOnly</code>, but WebLogic Builder was still populating the entire combo box from scratch.</p> <p>In addition, the <code>EJBUtils.getCMPFields()</code> method returned not only CMP fields, but also CMR fields. Therefore, the nodes appeared to have random values in them, because they were getting accidentally set to a value that was actually a CMR field. Customers who experienced these problems would also notice a corrupted <code>ejb-jar.xml</code> file.</p> <p>The code now determines if users are adding a new CMP field or editing an existing one, and only populates the combo box when users are adding a new one. In addition, a filter method now filters out CMR fields. As a result of these changes, <code>ejb-jar.xml</code> files will no longer get corrupted.</p>	8.1 SP2
CR108648	<p>Running the WebLogic EJB-to-JSP integration tool on the command line resulted in a <code>NoClassDefFound</code> error. In addition, this tool only worked for 1.0 EJBs.</p> <p>The class <code>weblogic.servlet.ejb2jsp.gui.Main</code> was added to prevent the <code>NoClassDefFound</code> error, and code was added to handle 2.0 EJBs. Now, if an EJB has ONLY a <code>Local/LocalHome</code> interface, the tool reads the methods from that interface pair. If an EJB has ONLY a <code>Remote/RemoteHome</code> interface, the tool reads the methods from that interface pair. If an EJB has BOTH interface pairs, the tool reads the methods from the <code>Local/LocalHome</code> interface.</p>	8.1 SP2

## Web Services

Change Request Number	Description	Release Fixed
CR086326	The JSSEAdapter was pending a change to JDK 1.4, and therefore was not complete. The JSSEAdapter has been brought up to parity with the WLSAdapter for strict checking, verbose and localIdentity features. As a result of this change, the JSSEAdapter now provides the same functionality as the WLSAdapter.	8.1 SP2
CR092268	A client running behind a firewall needed to set properties on a Web service stub. The problem was resolved by the introduction of two new properties, <code>weblogic.webservice.client.proxyusername</code> and <code>weblogic.webservice.client.proxypassword</code> .	7.0 SP4 8.1 SP2
CR103346	The Web Service test page did not support the <code>documentwrapped</code> invocation style. Support for testing the <code>documentwrapped</code> invocation style was added to the Web Service test page.	8.1 SP2
CR103512	Some properties set on the stub were not copied into the <code>MessageContext</code> .	7.0 SP4
CR107171	After a code change, WebLogic Server now supports the following usecases: Endpoint can be modified in the following flow: <ul style="list-style-type: none"> <li>• <code>Stub-&gt;MessageContext in handleRequest-&gt;MessageContext in handleResponse-&gt;StubCall-&gt;MessageContext in handleRequest-&gt;Call</code></li> </ul> Timeout can be set in the following order: <ul style="list-style-type: none"> <li>• <code>Stub-&gt;MessageContext in handleRequest</code></li> <li>• <code>Call-&gt;MessageContext in handleRequest</code></li> </ul>	8.1 SP2
CR104199	If you used a .NET C# client to access a WebLogic Server Web service and requested two strings returned, the first a result and the second an in/out parameter, the order of the strings returned was incorrect. The second string returned as the result, and the in/out parameter returned as null. A code change ensures that the first accessor returned is the return value.	7.0 SP4 8.1 SP2
CR104276	The command-line flag <code>-Dweblogic.webservice.verbose=true</code> was causing an error message during startup, saying that it is not recognized by WebLogic Server. This error message was wrong, since the flag was being recognized by WebLogic Server. This message has been removed.	8.1 SP2

Change Request Number	Description	Release Fixed
CR104481	<p>A Web Service whose method contains multi-byte characters could not be accessed from the testing page in the Administration Console. (The client application can, however, access the Web Service.)</p> <p>This problem occurred because a request parameter was being parsed using ISO-8859-1 encoding. It was resolved by modifying the code to set a character encoding before parsing a request parameter.</p>	8.1 SP2
CR105140	<p><code>HttpRequest</code> and <code>HttpResponse</code> not available in the server handler's message context. This was a regression from WebLogic Server 7.0.</p> <p>This problem has been resolved by adding support for <code>MessageContext.getProperty("HttpRequest")</code> and <code>MessageContext.getProperty("HttpResponse")</code> in the server handler.</p> <p><b>Note:</b> <code>HttpRequest</code> and <code>HttpResponse</code> are now deprecated. Use <code>weblogic.webservice.transport.http.request</code> and <code>weblogic.webservice.transport.http.response</code> instead.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR106392	<p>Running with <code>invocation-style="one-way"</code> resulted in the WSDL containing an <code>&lt;output&gt;</code> in the <code>&lt;binding&gt;</code> section for that operation as can be seen from the snippet below.</p> <pre data-bbox="229 522 989 1124"> &lt;portType name="MyServicePort"&gt;   &lt;operation name="sayNothing"&gt;     &lt;input message="tns:sayNothing" /&gt;   &lt;/operation&gt; &lt;/portType&gt; &lt;binding name="MyServicePortSoapBinding" type="tns:MyServicePort"&gt;   &lt;soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"; /&gt;   &lt;operation name="sayNothing"&gt;     &lt;soap:operation soapAction="" style="document" /&gt;     &lt;input&gt;       &lt;soap:body use="literal" namespace="http://tempuri.org/"; /&gt;     &lt;/input&gt;     &lt;output&gt;       &lt;soap:body use="literal" namespace="http://tempuri.org/"; /&gt;     &lt;/output&gt;   &lt;/operation&gt; &lt;/binding&gt; </pre> <p>This behavior was exhibited in both "rpc" and "document" style Web services. Running with Microsoft's WSDL . EXE against this WSDL resulted in an unsuccessful validation and consequently in a failure to generate the .Net client proxy or stubs.</p> <p>A code change stopped the generation of output messages in one-way bindings, resolving the problem.</p>	7.0 SP3 8.1 SP2
CR107312	<p>Clientgen-generated clients wrote attributes in the SOAP request without namespaces, even if <code>attributeFormDefault="qualified"</code> was specified. WebLogic Workshop Web Services did not recognize these attributes, causing the corresponding field of the Java object to be set to null.</p> <p>Namespaces of attributes are now written out if <code>attributeFormDefault="qualified"</code>.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR108646	<p>The servicegen task did not support the "mergewithexistingws" attribute, resulting in a build failure.</p> <p>This problem occurred because no <code>setMergeWithExistingWS</code> method was defined in the <code>weblogic.ant.taskdefs.webservices.servicegen.ServiceGenTask</code> class.</p> <p>The following method is now part of the <code>ServiceGenTask</code> class:</p> <pre>public void setMergeWithExistingWS(boolean b) { mergeWithExistingWS = b; }</pre>	8.1 SP2
CR109437	<p>When casting an <code>SSLAdapter</code> to a <code>WLSSLAdapter</code> at runtime as shown below, a <code>ClassCastException</code> was thrown.</p> <pre>SSLAdapterFactory factory = SSLAdapterFactory.getDefaultFactory(); WLSSLAdapter adapter = (WLSSLAdapter) factory.getSSLAdapter();</pre> <p>This problem was caused by an incomplete feature addition.</p> <p>The <code>JSSEAdapter</code> was correctly entered secondary to the default <code>WLSSLAdapter</code>. This fix restored the original behavior.</p>	8.1 SP2
CR109624	<p>Exceptions were always being printed to <code>stdout</code> from Web Service clients.</p> <p>Logging on the client side now only happens if the system property <code>weblogic.webservice.client.logging</code> is set to <code>true</code>. The default is <code>false</code>.</p>	8.1 SP2
CR109898 CR112443	<p>Passing <code>org.w3c.dom.Document[]</code> as a parameter to a method causes a <code>SOAPException</code> beginning with the following lines:</p> <pre>javax.xml.soap.SOAPException: Found SOAPElement [&lt;anyType&gt; &lt;this xmlns="mynamespace"&gt; &lt;f:that xmlns:f="yournamespace"&gt; &lt;or&gt; a lot of random &amp;lt; &lt;/or&gt; &lt;f:the&gt; &lt;/f:the&gt; &lt;f:other&gt; foo bar blaz&lt;/f:other&gt; &lt;/f:that&gt; &lt;/this&gt; &lt;/anyType&gt;]. But was not able to find a Part that is registered with this Message which corresponds to this SOAPElement ...</pre> <p>A code fix resolved this problem.</p>	8.1 SP2
CR109938	<p>When the Web Service runtime attempted to find out the encoding of the XML represented by the Source object, its underlying stream was read and not reset.</p> <p>The XML stream is now reset.</p>	8.1 SP2

Resolved Problems for Service Pack 2

Change Request Number	Description	Release Fixed
CR110168	<p>Clients failed with an <code>IllegalArgumentException</code> if the XML returned from the server did not explicitly indicate the size of a SOAP array.</p> <p>This problem occurred because the argument used for creating a new array instance was null because of missing size information.</p> <p>For a SOAP array of an unspecified size, objects are now stored in a list first, and then the array is created.</p>	7.0 SP4 8.1 SP2
CR110945	<p>A document or literal Web Service involving arrays generated a SOAP response message whereby the namespaces were not being applied properly. This resulted in .Net clients to ignore some of the data. WebLogic Server ant tasks (<code>autotype</code>, <code>wsdl2service</code>, and <code>wspackage</code>) are used to generate the Web Service.</p> <p>This problem was resolved by modifying the code to write proper namespaces to the generated codec.</p>	8.1 SP2
CR111163	<p>When iterating through the child nodes of the SOAP header, empty text nodes were casted to <code>SOAPHeaderElement</code>'s.</p> <p>Text nodes are now discarded. However, there was another change required to make the <code>examineHeaderElements()</code> method work. The <code>mustUnderstand</code> and <code>actor</code> attributes are now being recognized with and without namespace qualification. Previously, only the non-qualified form was recognized.</p>	8.1 SP2
CR111187	<p>In the <code>weblogic.webservice.binding.soap.HttpResponse</code> class, the method <code>getBodyAsString</code> threw a "String index out of range" error.</p> <p>A code change causes the method to use the length of the body, instead of the end, resolving the problem.</p>	7.0 SP4 8.1 SP2
CR111393	<p>A <code>NullPointerException</code> occurred during deployment of a simple EAR file using WebLogic Builder.</p> <p>This problem has been resolved.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR109033 CR111653	<p>Customers could not specify a <code>KeyWrappingMethod</code> for encryption, then configure the data security for Web Services. (By default, only <code>rsa-oaep-mgflp</code> was used.)</p> <p>Customers can now specify any supported <code>KeyWrappingMethod</code> in the deployment descriptor for a Web Service. For example:</p> <pre>&lt;spec:EncryptionSpec EncryptionMethod="http://www.w3.org/2001/04/xmlenc#tripleDES-cbc" KeyWrappingMethod="http://www.w3.org/2001/04/xmlenc#rsa-1_5" EncryptBody="true"&gt; &lt;/spec:EncryptionSpec&gt;</pre>	8.1 SP2
CR111881 CR128662	<p>WebLogic Server did not provide a timeout to a Web Service invocation on the client-side.</p> <p>To timeout a Web Service invocation on the client-side in WebLogic Server SP 2:</p> <ol style="list-style-type: none"> <li>1. Make sure <code>weblogic.jar</code> is available on the client side.</li> <li>2. Set the <code>weblogic.webservice.UseWebLogicURLStreamHandler</code> system property to <code>true</code>.</li> <li>3. Set the timeout on the stub as follows: <ul style="list-style-type: none"> <li>• With the API for <code>BindingInfo:BindingInfo bInfo = (BindingInfo) stub._getProperty("weblogic.webservice.bindinginfo"); bInfo.setTimeout(5 /* secs */);</code></li> <li>• With the stub property: <pre>stub._setProperty("weblogic.webservice.rpc.timeoutsecs", "5" /* secs */);</pre> </li> </ul> </li> </ol> <p><b>Note:</b> In WebLogic Server 8.1 SP1, <code>BindingInfo.setTimeout(5000/*millisecs*/) took milliseconds as the parameter. However, this had no effect on the client side. With SP 2, the parameter takes seconds instead of milliseconds.</code></p>	8.1 SP2
CR112835	<p>When using <code>wsdl2service</code> to generate a service for mixed-style WSDL, the client side threw a <code>SerializationException</code> while invoking the service.</p> <p>Mixed style is only supported on the client-side but not on the server-side. As a result of this change, <code>wsdl2service</code> will no longer generate a service if it is mixed style. Instead, a <code>WSBuildException</code> will be thrown while running <code>wsdl2service</code>.</p> <p><b>Note:</b> Prior to this change, <code>wsdl2service</code> would generate a mixed-style service.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR113038	<p>WebLogic Server 8.1 supported an <a href="#">old version of the SOAP 1.2 specification</a>.            In WebLogic Server 8.1 SP2, support for the <a href="#">latest SOAP1.2 specification</a> has been added.</p>	8.1 SP2
CR113082	<p>A <code>java.lang.Object</code> could not be used as a parameter or return type of a Web Service. As a result, an error similar to the following occurred during compilation:</p> <pre>Client.java:79: inconvertible types found : void required: java.lang.Object java.lang.Object retVal = (java.lang.Object)port.mapbasic8Data(arg0);</pre> <p>A code fix resolved the problem.</p>	8.1 SP2
CR113095	<p>While calling the <code>getContent()</code> method in the <code>handleResponse()</code> method of a server-handler chain as follows:</p> <pre>SOAPMessage msg = (SOAPMessageContext)context ).getMessage(); javax.xml.transform.Source src = msg.getSOAPPart().getContent();</pre> <p>an exception starting with the following lines was thrown:</p> <pre>org.w3c.dom.DOMException: NAMESPACE_ERR: An attempt is made to create or change an object in a way which is incorrect with regard to namespaces....</pre> <p>This exception has been eliminated, and the correct <code>javax.xml.transform.Source</code> object will now be returned when invoking <code>msg.getSOAPPart().getContent()</code>.</p>	8.1 SP2
CR120422	<p>The following classes were missing from <code>webserviceclient.jar</code>:</p> <pre>weblogic/webservice/encoding/AttachmentCodec.class weblogic/webservice/encoding/DataHandlerArrayCodec.class weblogic/webservice/encoding/DataHandlerCodec.class weblogic/webservice/encoding/ImageArrayCodec.class weblogic/webservice/encoding/ImageCodec.class weblogic/webservice/encoding/MimeMultipartArrayCodec.class weblogic/webservice/encoding/MimeMultipartCodec.class weblogic/webservice/encoding/XMLSourceArrayCodec.class weblogic/webservice/encoding/XMLSourceCodec.class</pre> <p>The missing classes have been added.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR120835	<p>A class required to process client-side JAX-RPC handlers (<code>javax.xml.rpc.handler.GenericHandler</code>) was not included in the <code>webserviceclient.jar</code> file that ships with WebLogic Server.</p> <p>The required class was added to <code>webserviceclient.jar</code>. Now Web Service clients can use JAX-RPC handlers using only <code>webserviceclient.jar</code>, and it is not necessary to include <code>webservices.jar</code> in the client's classpath.</p>	8.1 SP2
CR120957	<p>Static clients did not throw the correct exception when multiple exceptions are thrown from a EJB method. For example, a static client for the method signature:</p> <pre data-bbox="292 699 1110 722">public void throwExceptions() throws ExceptionA, ExceptionB</pre> <p>always threw <code>ExceptionA</code>, regardless of what exception had been thrown from the server side.</p> <p>The static client now throws the exception which matches the one thrown from server-side.</p>	8.1 SP2
CR121013	<p>If an authenticated subject has been set on the message context, it is now used for invoking the backend component.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR121088	<p>In WebLogic Server 8.1 SP1, an MDB is published as a Web Service using <code>servicegen</code>. This MDB listens on a foreign JMS provider (MQ-Series 5.3). Testing of the Web Service through the Administration Console, a <code>ClassCastException</code> starting with the following lines was raised:</p> <pre>Invocation failed ! Parameter Name Parameter Value Request sent to the server &lt;!--REQUEST.....--&gt; &lt;env:Envelope   xmlns:env="http://schemas.xmlsoap.org/soap/envelope/";   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance";   xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/";   xmlns:xsd="http://www.w3.org/2001/XMLSchema";&gt; &lt;env:Header&gt; &lt;/env:Header&gt; &lt;env:Body&gt;   &lt;n1:toupper xmlns:n1="http://localhost:7001/messageToupperWebservices"; sample string&lt;/n1:toupper&gt; &lt;/env:Body&gt; &lt;/env:Envelope&gt;</pre> <p>Response from the server</p> <pre>&lt;!--RESPONSE.....--&gt; &lt;env:Envelope   xmlns:env="http://schemas.xmlsoap.org/soap/envelope/";   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance";   xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/";   xmlns:xsd="http://www.w3.org/2001/XMLSchema";&gt; ...</pre> <p>The <code>ClassCastException</code> will no longer be raised; a code fix resolved the problem.</p>	8.1 SP2
CR122437	<p>The <code>InputStream</code> field of <code>StreamSource</code> was always used, even if it was null. If the <code>InputStream</code> is null, the <code>Reader</code> is now used. If both are null, a <code>SOAPException</code> is thrown.</p>	8.1 SP2
CR124019	<p><code>Clientgen</code> failed to compile a generated class for an extended type if the base type had a different namespace.</p> <p>This problem occurred because base classes from different packages were not being imported.</p> <p>Base class names are now generated with complete package names.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR124285	<p>Clientgen failed with a <code>ClassNotFoundException</code> on WSDL with references to types in another schema, which had no <code>targetNamespace</code> via the generic <code>&lt;xs:import&gt;</code> tag.</p> <p>A code fix resolved this problem. The fix also requires <code>&lt;clientgen&gt;</code> to define <code>typePackageName</code> or <code>typePackageBase</code> if there is no <code>targetNamespace</code> in the schema.</p>	8.1 SP2
CR127338	<p>WebLogic Server generated a MIME header with the string <code>"start=null"</code>. This violated RFC2387 in which <code>"start element"</code> is optional but can not be null. For example:</p> <pre data-bbox="292 701 1123 951"> [java] Content-Type=multipart/related;boundary="====_Part_0_195615 02.106 8580828000";start=null [java] Transfer-Encoding=Chunked [java] Envelope : [java] [java] =====_Part_0_19561502.1068580828000 [java] Content-Type: text/xml; charset='utf-8' [java] Content-Transfer-Encoding: 8bit </pre> <p>This problem was resolved by modifying WebLogic Server to generate the correct <code>"start element"</code>. For example,</p> <pre data-bbox="292 1031 1123 1274"> [java] Content-Type=multipart/related;boundary="====_Part_0_296080 4.1068582126531";start=__WLS__1068582126546__SOAP__ [java] Transfer-Encoding=Chunked [java] Envelope : [java] [java] =====_Part_0_2960804.1068582126531 [java] Content-Type: text/xml; charset='utf-8' [java] Content-Transfer-Encoding: 8bit [java] Content-ID: __WLS__1068582126546__SOAP__ </pre>	8.1 SP2

## WTC

Change Request Number	Description	Release Fixed
CR070627	<p>The WebLogic Server Administration Console did not check the format for the Network Address field. This problem occurred for either WTC LocalTuxDom or RemoteTuxDom configurations.</p> <p>This problem was resolved by updating <code>WTCLegalHelper</code> to include a method for checking that the format of a network address is valid. As a result of this change, an error will now be reported when an invalid format is used to define the Network Address field.</p>	8.1 SP2
CR107333	<p>The design for configuring the View classes to be used by WTC was expected to be only through the JMX interface. This has proven to be an unwieldy usage for WebLogic Workshop with Tuxedo-controls.</p> <p>A method has been added to the <code>TuxedoConnection</code> that permits customers to append the WTC View HashTable with a View class they have loaded. Now customers do not need to update the <code>WTCResourcesMBean</code> with the View class they will be using when a <code>WTCService</code> is running. Customer applications may load the class and then call the <code>updateViewMap</code> method on the <code>TuxedoConnection</code>. Side-effects are that the MBean information is not updated and the View HashTable is open to appends from non-administrators.</p>	8.1 SP2
CR109849	<p>WTC uses a separate protocol for communication with Tuxedo 6.5 domains. One part of this communication is creating a <code>TypedBuffer</code> to receive the data that is read in. In this case, a view buffer must be created, and this requires that the view subtype be provided. (The subtype is the name of a view which maps to a class listed in the Resources configuration section.) However, no subtype was being given to the <code>createTypedBuffer</code> method.</p> <p>The subtype value is now read in from the input stream and passed to the <code>createTypedBuffer</code> method. As a result, WTC 8.1 will now process View buffers from Tuxedo 6.5 domains.</p>	8.1 SP2
CR109877	<p>The <code>TypedFML.FldId()</code> method was returning a null value instead of throwing a <code>Error</code> exception as expected.</p> <p>A throw statement was added to <code>TypedFML.FldId()</code> for the case where no field ID is found. As a result of this change, when no field ID is found that corresponds to the input name parameter, an <code>Error.FBADNAME</code> exception will be thrown.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR110488	The <code>wsdl2service</code> task had the following problems handling user-defined exceptions:	8.1 SP2
CR110619	<ul style="list-style-type: none"> <li>• Operations of the interface did not throw the user-defined exception as specified in the WSDL.</li> <li>• There were no fault entries in the deployment descriptor <code>web-services.xml</code>.</li> </ul> <p>This issue has been resolved. As a result, <code>wsdl2service</code> now handles user-defined exception properly.</p>	
CR110957	<p>Web Service applications had problems with VIEW buffer fields defined as strings. While these fields are intended to hold null-terminated character strings, Tuxedo actually creates them as fixed length character arrays. Tuxedo sends the entire array as part of a VIEW buffer. When WTC receives such a buffer, it converts the entire array to a Java byte array and then to a Java <code>String</code>. The resulting string may contain many '0' characters or garbage characters at the end of the real data. WTC should have searched the incoming character array for the first null byte and truncated the byte array it created at that point.</p> <p>A new option was added to the <code>viewj</code> and <code>viewj32</code> compilers: <code>-modify_strings</code>. Additionally, new versions of the <code>xdr_encode_string</code> and <code>xdr_decode_string</code> methods of the <code>wtc.jatmi.Utilities</code> class were added. If <code>-modify_strings</code> is specified, the compilers will generate code to use the new versions of the string encode and decode routines. These versions will truncate strings received from Tuxedo at the first null character and add a null character to any string sent to Tuxedo. No behavior changes unless the new <code>-modify_strings</code> option is given.</p>	8.1 SP2

Change Request Number	Description	Release Fixed
CR122200	<p>Viewj and Viewj32 now generate code in the field setters and getters that will set and use the values of the associated length and count fields, if the view definition specifies the L or C flags for a given field.</p> <ul style="list-style-type: none"> <li>The setters will set the count for a field and the length for a string or carray field.</li> <li>The getters for array fields will return an array that is at most the size of the associated count field.</li> <li>The getters for a string or carray will return data that is at most the length of the associated length field.</li> </ul> <p>This behavior can be controlled both at the time the TypedView or TypedView32 class is generated by viewj or viewj32, and at run-time.</p> <p>Two methods control the run-time behavior:</p> <ol style="list-style-type: none"> <li><code>boolean getAssociatedFieldHandling()</code> returns <code>true</code> if the current state is set so that setters and getters will use the associated length and count fields; otherwise it returns <code>false</code>.</li> <li><code>void setAssociatedFieldHandling(boolean state)</code> sets the state of associated field handling. If set to <code>false</code>, the setters and getters ignore the length and count fields. If set to <code>true</code>, they set and use the associated fields as described above.</li> </ol> <p>The default state for associated field handling is <code>false</code>. If the <code>-associated_fields</code> option is given to viewj or viewj32, the default state is set to <code>true</code>.</p>	8.1 SP2
CR122953	<p>The <code>Fchg(FmlKey key, Object value)</code> method in the <code>weblogic.wtc.jatmi.TypedFML32</code> class did not correctly fill all prior entries with null values as documented.</p> <p>This problem was resolved with a code fix.</p>	7.0 8.1 SP2

# XML

---

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR120910	<p>In web.xml, if "MS950" or "CP950" was specified as the encoding name, an exception starting with the following lines was thrown:</p> <pre>weblogic.management.ApplicationException: Exception:weblogic.management.ApplicationException: Prepare failed. Task Id = 21 Module: encoding Error: [HTTP:101062][HTTP] Error reading Web application "C:\bea811\user_projects\domains\mydomain\.\myserver\upload\ encoding\encoding.war". java.io.UnsupportedEncodingException: [J2EE:160098]Unable to parse the file 'C:\bea811\user_projects\domains\mydomain\.\myserver\upload\ encoding\encoding.war:WEB-INF/web.xml' because it uses an invalid encoding name, 'Cp950'. All deployment descriptors must use an IANA or Java encoding name. Please change the encoding of the descriptor to be a supported encoding. ...</pre> <p>A code change fixed the above exception, since "MS950" and "CP950" are valid Java encoding names.</p>	8.1 SP2

---

## Resolved Problems for Service Pack 2

# Resolved Problems for Service Pack 1

Service Packs are cumulative; Service Pack 1 contains all the fixes made in earlier Service Packs released for WebLogic Server 8.1.

The following sections describe problems that were resolved for the release of WebLogic Server 8.1 Service Pack 1.

- [“Administration Console” on page 6-3](#)
- [“Classloader” on page 6-8](#)
- [“Cluster” on page 6-9](#)
- [“Connector” on page 6-9](#)
- [“CORE” on page 6-13](#)
- [“Deployment” on page 6-17](#)
- [“EJB” on page 6-19](#)
- [“Installation” on page 6-28](#)
- [“Internationalization” on page 6-29](#)
- [“J2EE” on page 6-30](#)
- [“JDBC” on page 6-31](#)
- [“JMS” on page 6-36](#)

## Resolved Problems for Service Pack 1

- “JNDI” on page 6-42
- “JTA” on page 6-43
- “JVM” on page 6-46
- “Localization” on page 6-46
- “Node Manager” on page 6-47
- “Plug-Ins” on page 6-47
- “RMI/RMI-IIOP” on page 6-49
- “Security” on page 6-51
- “Servlets & JSPs” on page 6-54
- “System Administration” on page 6-64
- “Tools” on page 6-70
- “Utilities” on page 6-71
- “WebLogic Workshop” on page 6-72
- “Web Services” on page 6-73
- “XML” on page 6-78

# Administration Console

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR094513	For Connectors that had not yet been used, the Connector monitoring table displayed the last used time as “Wed Dec 31 19:00:00 EST 1969.” It now displays the text: “Never.”	8.1 SP1
CR095863	Custom attributes defined in the MBean Definition Files (MDFs) for several WebLogic security providers were displayed on the General tab for that provider, rather than on the Details tab. These attributes were as follows: <ul style="list-style-type: none"><li>• RequireUnanimousPermit for the WebLogic Adjudication provider</li><li>• Severity for the WebLogic Auditing provider</li><li>• MinimumPasswordLength of the WebLogic Authentication provider</li><li>• PolicyDeploymentEnabled for the WebLogic Authorization provider</li><li>• CredentialMappingDeploymentEnabled for the WebLogic Credential Mapping provider</li><li>• RoleDeploymentEnabled for the WebLogic Role Mapping provider</li></ul> These attributes are now on the Details tabs for these WebLogic security providers.	8.1 SP1
CR097818	No online help was available for the View Idle Connections or View Leaked Connections pages. Online help for these pages is now available.	8.1 SP1
CR098353	The TransactionLogFileWritePolicy attribute was added in 7.0 SP01, but was never exposed in the Administration Console. A field for setting this attribute is now available on the Server →Logging →JTA tab.	7.0 SP3 8.1 SP1
CR099541	The Delete button at the bottom of the Policy Editor page did not delete the highlighted security policy. It also did not delete the default security policy. However, the Remove button would delete the default security policy.  This action occurred because of a problem in the JavaScript code.  The JavaScript code was modified so that the buttons on the Policy Editor page now work properly.	8.1 SP1

Change Request Number	Description	Release Fixed
CR099771	<p>The JDBC Assistant did not allow users to enable "Advanced Options" when creating data sources. The particular option of interest was 'Emulate Two Phase Commit for non XA Driver' .</p> <p>The 'Emulate Two Phase Commit for non XA Driver' option is now available in the JDBC Assistant.</p>	8.1 SP1
CR099866	<p>If the administration port is enabled, the Administration Console must use it. However, console extension Web applications do not use the administration port, so the links in the navigation tree that pointed to the console extension Web application did not work.</p> <p>This problem occurred because console extension Web applications were not labeled internal (which is required to access the administration port), and thus the servlet container rejected the request.</p> <p>Console extension Web applications are now labeled as internal.</p>	8.1 SP1
CR100224	<p>Exceptions were not correctly handled when viewing the JNDI tree for a Managed Server. The exceptions showed up in the log, but there was no indication in the Administration Console that an exception had occurred or that anything out of the ordinary had happened.</p> <p>The JSP code has been modified so that when an exception in the JNDI tree occurs, an "Access Denied" entry for the node in question is made in the navigation tree. The exception is also printed to the log.</p>	8.1 SP1
CR100460	<p>When an attempt was made to clone a server that is a target of an existing JMS distributed queue/topic, a <code>javax.management.InvalidAttributeValueException</code> was displayed in the Administration Console. This exception stated that the server was an "Illegal target" and that "A <code>JMSDistributedQueue/JMSDistributedTopic</code>, if it has members, should be deployed to a cluster or a single server that is not in a cluster." While the newly created/clone server was created, it did not show up in the navigation tree, but did show in the summary server view.</p> <p>Code has been modified so that the cloning of a server would not cause existing JMS distributed queues/topics to be added as targets, since these have already been targeted to a cluster. Additionally, the newly created/clone server now shows up in the navigation tree.</p>	8.1 SP1

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR100638	<p>Clicking the Distributed Topic Members or Distributed Queue Members links in the banner of the page for a specific Distributed Topic/Queue Member resulted in a 404 error.</p> <p>This problem was resolved by removing the links from the banner, since they duplicated the functionality of the links for the specific Distributed Topic/Queue Member.</p>	8.1 SP1
CR100731	<p>After the apparently successful deletion of a Web application, clicking the Continue link caused a <code>javax.management.InstanceNotFoundException</code> to be displayed.</p> <p>This problem was resolved with a code fix.</p>	8.1 SP1
CR100974	<p>When configuring a new Network Channel for a server, the Administration Console required that a Listen Address be specified, when this field is not required.</p> <p>This problem was resolved by removing validation on the Listen Address field so that it is not required by the Administration Console.</p>	8.1 SP1
CR101247	<p>If you were logged in to the Administration Console as a user who was granted a non-Admin security role and clicked the Services →COM node in the navigation tree, the Domain Configuration page was displayed.</p> <p>This problem was resolved by removing the Services →COM node from the navigation tree when users do not have permission to modify security settings (that is, for all users granted a non-Admin security role).</p>	8.1 SP1
CR101317	<p>When using the Administration Console with Internet Explorer 5.5, you may have seen the following JavaScript error message: "Object does not support this property or method". Upgrading to Internet Explorer 6.0 corrects this problem. This error appeared only occasionally on some computers where unspecified Internet Explorer patches have been applied.</p> <p>This problem was resolved by adding a JavaScript workaround for the bug in this version of Internet Explorer.</p>	8.1 SP1
CR101770	<p>The SSL Login Timeout field was unintentionally dropped from the WebLogic Server Administration Console.</p> <p>This problem was resolved by adding this field to the Configuration →Tuning tab, under the Login Timeout field.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR102441	<p>There was no help text displayed on the pages that allow you to select which configured Authentication provider, Authorization provider, Role Mapping provider, or Credential Mapping provider should store a new user, group, security policy, or security role that was being defined (respectively).</p> <p>This problem was resolved by adding and displaying help text for these pages.</p>	8.1 SP1
CR102561	<p>When a JMS file store was assigned to a single JMS Server, and a distributed topic was created with the Enable Store box checked (that is, set to true), errors similar to the following may have been displayed:</p> <p>StoreEnabled set to 'true' for destination 'MyJMS Topic' but no Store configured for server 'TendrilmJMSserver'.</p> <p>Can not set Enable Store to True if the JMS Server does not have a store defined.</p> <p>This problem occurred because the Enable Store setting is non-dynamic and the JMS Server was being auto-targeted too early in the process.</p> <p>This problem was resolved by removing the auto-targeting of JMS Servers.</p>	8.1 SP1
CR102846	<p>The Administration Console displayed a Channel Weight field in the Server <del>—</del>Protocol <del>—</del>Channels <del>—</del>Network Channel tab's Advanced Options pane, when this field was obsolete.</p> <p>This problem was resolved by removing the Channel Weight field from the Administration Console.</p>	8.1 SP1
CR102982	<p>The Client Cert Proxy Enabled check box was displayed on the Tuning tab for a server, but this attribute is not related to tuning.</p> <p>This problem was resolved by relocating the ClientCertProxyEnabled field to the Configuration <del>—</del>General tab for a server.</p>	7.0 SP3 8.1 SP1

Change Request Number	Description	Release Fixed
CR103328	<p>When attempting to view server log message details using the Administration Console, the following exception may have occurred:</p> <pre>java.lang.IllegalArgumentException: Cannot format given Object as a Number</pre> <p>This problem occurred because messages with number arguments were not being handled properly.</p> <p>This problem was resolved by modifying the code so that messages with number arguments were handled properly. In addition, unexpected exceptions that may have occurred while they were being formatted were also fixed. A localized message will now be presented to users as happens elsewhere in the Administration Console.</p>	8.1 SP1
CR103536	<p>When multiple JMS Servers were configured and new distributed topics/queues were created with the Enable Store box unchecked (that is, set to false) and then later checked (that is, set to true), an error similar to the following was displayed:</p> <pre>Can not set Enable Store to True if the JMS Server does not have a store defined.</pre> <p>This error also occurred if the Enable Store box was checked, a new topic/queue was prepared, and the Apply button was clicked after the topic/queue was created (by clicking the Create button).</p> <p>This problem was resolved by correcting the verification logic.</p>	8.1 SP1
CR105226	<p>When attempting to view global roles without use of the navigation tree (by clicking <code>myrealm</code> → <code>User Management</code> → <code>Manage Role</code> within this Security Realm), the resulting page displayed all security roles, not just the global roles. This may have resulted in seemingly duplicate security roles, when in fact a global role and scoped role with the same name may have been listed.</p> <p>This problem was resolved by modifying the page so that the navigation path shown above displays only global roles.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR105262	<p>Attempting to view scoped roles for a Web application deployed as part of an enterprise application (EAR) resulted in the following error:</p> <p>There are no appropriate RoleEditor providers configured.</p> <p>This problem was resolved by modifying the code so that it locates a Role Mapping provider that supports the editing, deploying, and so on of security roles by implementing the RoleEditor optional SSPI.</p>	8.1 SP1
CR108134	<p>The JDBC Assistant listed a number of JDBC drivers, but only a subset of those drivers are actually certified for WebLogic Server JDBC.</p> <p>The drivers that are certified for WebLogic Server JDBC are now marked as such in the JDBC Assistant.</p>	8.1 SP1

## Classloader

Change Request Number	Description	Release Fixed
CR102530	<p>When WebLogic Server executes external processes that require CLASSPATH information, a String representation is generated based on the classpath of the current classloader. This representation includes information from any MANIFEST classpath entries and could have caused duplicate elements in the generated path. In some cases, this might have led to an OutOfMemoryError.</p> <p>The algorithm that generates the classpath has been modified to filter duplicate entries in order to prevent this problem.</p>	8.1 SP1

## Cluster

Change Request Number	Description	Release Fixed
CR102526	<p>When an HTTP network channel was defined on a server (using the Administration Console's Protocols →Channels tab for that server), session replication did not work.</p> <p>This problem occurred because there was a problem in the remote channel selection code.</p> <p>This problem was resolved by modifying the remote channel selection code so that it does not try to use invalid channels for t3.</p>	8.1 SP1

## Connector

Change Request Number	Description	Release Fixed
CR100269	<p>JDBC Resource Adapter using Oracle Thin Client driver failed during completion of client-controlled transactions.</p> <p>This problem occurred when using a transaction started from a client application. The client started a transaction and then accessed a bean or Web application that used the resource adapter. When the client application attempted to commit or roll back the transaction, the server threw an exception of type <code>javax.transaction.xa.XAException</code>.</p> <p>This problem occurred because the Oracle Thin Client driver did not properly handle a call to <code>XAResource.end</code> with a <code>TMSUSPEND</code> flag.</p> <p>This problem was resolved by having the connection intercept the calls to the <code>XAResource</code>.</p>	8.1 SP1
CR100707	When using JCA in versions of WebLogic Server prior to 7.0 SP3, a <code>java.lang.ClassCastException</code> with	7.0 SP3
CR101761	<code>java.naming.CompositeName</code> was thrown. When using JCA in WebLogic Server 8.1, <code>resource-ref</code> entries (Auth, SharingScope) returned <code>null</code> for Web applications.	8.1 SP1
	This problem occurred because of an internal error, and was resolved with a code fix.	

Change Request Number	Description	Release Fixed
CR101906	<p>A client invoked a method on an EJB. When the connection failed between the ManagedConnection (MC) and the EIS while in XA transaction state, the container called XAResource.rollback and ManagedConnection.cleanup every 60 seconds after the transaction had already been rolled back. Subsequent calls made to the EIS resulted in a java.lang.NullPointerException exception:</p> <pre data-bbox="266 579 1036 1454"> &lt;Mar 21, 2003 5:50:52 PM PST&gt; &lt;Error&gt; &lt;Connector&gt; &lt;190041&gt; &lt;&lt; JCA Resource Adapter for ClearPath MCP_eis/comsRAJNDINAMEBrazil &gt; The returned ManagedConnection instance is null&gt; &lt;Mar 21, 2003 5:50:52 PM PST&gt; &lt;Info&gt; &lt;EJB&gt; &lt;010051&gt; &lt;EJB Exception during invocation from home: drummer.CedarBank.CedarBankBeanBrazil_ris6bf_HomeImpl@12 fc36 threw exception: java.lang.NullPointerException java.lang.NullPointerException at drummer.CedarBank.CedarBankBeanBrazil.cedarBank(CedarBan kBeanBrazil.java:130) at drummer.CedarBank.CedarBankBeanBrazil_ris6bf_EOImpl.ceda rBank(CedarBankBeanBrazil_ris6bf_EOImpl.java:46) at drummer.CedarBank.CedarBankBeanBrazil_ris6bf_EOImpl_WLSk el.invoke(Unknown Source) at weblogic.rmi.internal.BasicServerRef.invoke(BasicServerR ef.java:362) at weblogic.rmi.cluster.ReplicaAwareServerRef.invoke(Replic aAwareServerRef.java:114) at weblogic.rmi.internal.BasicServerRef\$1.run(BasicServerRe f.java:313) at weblogic.security.service.SecurityServiceManager.runAs(S ecurityServiceManager.java:785) at weblogic.rmi.internal.BasicServerRef.handleRequest(Basic ServerRef.java:308) at weblogic.rmi.internal.BasicExecuteRequest.execute(BasicE xecuteRequest.java:30) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java :153) at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:134 ) </pre> <p>Research revealed that WebLogic Server failed to wait for the transaction to complete before trying to destroy the connection.</p> <p>A code fix resolved the problem.</p>	7.0 SP3 8.1 SP1

Change Request Number	Description	Release Fixed
CR101924	<p>When there is more than one matching credential, creating a new credential mapping resulted in an exception that began with the following lines:</p> <pre>java.lang.IllegalStateException: Subject is read-only at javax.security.auth.Subject\$SecureSet.add(Subject.java:1 046) at weblogic.connector.common.internal.SecurityContext\$1.run (SecurityContext.java:250) at java.security.AccessController.doPrivileged(Native Method) at weblogic.connector.common.internal.SecurityContext.inits ubject(SecurityContext.java:244) ... </pre> <p>This problem occurred because WebLogic Server called the <code>setReadOnly</code> method on the Subject as it added each credential in the list to the Subject. Therefore, only the first credential added to the Subject would not throw the above exception.</p> <p>This problem was resolved by modifying the code so that the <code>setReadOnly</code> method was called only once, after all the credentials were added to the Subject.</p>	8.1 SP1
CR102220	<p>A messaging bridge could not reconnect to the source/target destinations when the source Weblogic Server instance was killed and then restarted.</p> <p>This problem occurred because the <code>ConnectionPool</code> code was synchronized on the connection pool for the entire duration of releasing a resource or destroying a connection. This resulted in a deadlock of two threads, with the first thread releasing the resource and obtaining a lock, and then waiting on a second thread that was trying to destroy a resource that was waiting to get a lock on the connection pool.</p> <p>The problem was resolved by modifying the <code>ConnectionPool</code> code to obtain a lock only when it is required to remove any potential lock contention issues.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR104642	<p>When an error occurred in a <code>ManagedConnection</code>, it was removed from the connection pool, but the count of available connections in the pool was not being decremented. As a result, the maximum connection limit may have been reached sooner than expected, and an error message similar to the following may have been encountered when clients requested a new connection:</p> <pre data-bbox="266 574 969 652">&lt;Reached maximum capacity of pool "Archiv_eis/ArchivAdapter", making "0" new resource instances instead of "1".&gt;</pre> <p>This problem was resolved by modifying the code to properly decrement the number of available connections in the pool following an error in the <code>ManagedConnection</code>.</p>	8.1 SP1
CR105085	After doing a proxy test, the connection was not being closed.	8.1 SP1
CR105712	<p>This problem was resolved by modifying the code to call the <code>close()</code> method on the opened connection returned by <code>getConnection</code>. The connector container now introspects the connection handle to find the <code>close()</code> method and invokes it. If the <code>close()</code> method is not found, the container will log an error message stating that it did not find the <code>close()</code> method, and that it is required by the revised J2EE Connector Architecture Specification.</p>	
CR112051	<p>In some cases with transactions involving connections to a resource adapter, if one thread was in the middle of beginning a transaction, and another thread was completing a transaction, a deadlock situation may have occurred causing both threads to hang. The deadlock occurred because both the JTA and Connector subsystems were performing synchronization on the same object in a conflicting way.</p> <p>The internal synchronization mechanisms have been repaired so that the conflict will no longer occur. As a result, the server will no longer hang in cases where one thread is starting a transaction while at the same time another thread is completing a transaction.</p>	8.1 SP1

# CORE

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR071415	<p>When the WebLogic Server security manager starts and sets the policy file, the path <code>/usr/lib</code> was prepended to <code>javac</code> during JSP compilation, causing a <code>java.security.AccessControlException</code>.</p> <p>The problem was solved with a code fix.</p>	6.1 SP6 8.1 SP1
CR087808 CR095487	<p>When multiple initial context lookups were attempted from a client with different server URLs simultaneously, there was a possibility that WebLogic Server would shut down the healthy RJVM when detecting duplicate connections. This problem has been resolved.</p>	6.1 SP5 7.0 SP3 8.1 SP1
CR092155	<p>If remote calls are made to a stale stub (that is, a stub obtained from a previous incarnation of the server that can no longer be used to make remote calls), network resources may be depleted.</p> <p>This problem was resolved by modifying the code so that the communication layers detects this occurrence and throws a <code>ConnectException</code> without actually connecting to the server.</p>	8.1 SP1
CR097077	<p>When using a custom realm that made many outbound RMI calls, the reader threads became blocked by making outbound calls, leaving no threads for reading the response. This happens from the <code>BootServicesImpl.invoke</code> method from within the RJVM layer on a reader thread.</p> <p>This problem was resolved by moving <code>BootServicesImpl</code> into the RMI layer so that it can dispatch to the default execute queue.</p>	7.0 SP3 8.1 SP1
CR098865	<p>When SSL sockets were closed, they wrote data by sending a <code>CLOSE</code> alert to the peer. This caused threads to hang when clients became unresponsive or when the connection was very slow.</p> <p>This problem was resolved by ensuring that after <code>t3s</code> RJVM timeout, the socket will be forcibly closed.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR100177	<p>The use of <code>weblogic.security.service.ServerResource</code> was inconsistent with the way server resources were created in the Administration Console and with the documentation. When used to secure a T3 server, the name of the WebLogic resource was always "T3SRVR." In all other cases, the name of the server being acted upon was used as the name of the WebLogic resource.</p> <p>This problem was resolved by always using the name of the server being acted upon as the name of the WebLogic resource.</p>	7.0 SP3 8.1 SP1
CR101103	<p>WebLogic Server experienced a performance regression when running SPECjAppServer with IIOP.</p> <p>This problem occurred because the JDK IIOP serialization code queried the system class loader for two classes for every request, and because <code>loadClass</code> is synchronized. This made the lock highly contended across the whole VM.</p> <p>This problem was resolved by caching these two classes in <code>GenericClassLoader</code>.</p>	8.1 SP1
CR101720	<p>With connection filtering enabled, <code>Connection rejected</code> messages would accumulate in the log file, resulting in the log filling up.</p> <p>This was resolved with the new <code>ConnectionLogger</code> property, which lets users configure whether to log such messages to the log file.</p>	7.0 SP3 8.1 SP1
CR101946 CR101808	<p>If you used the Java socket reader implementation, extended use of JMS sometimes led to an Out of Memory error.</p> <p>This problem occurred because of a memory leak in the Java socket muxer, which happened when socket read errors occurred.</p> <p>The problem was resolved by modifying the code to ensure that data structures related to the sockets were freed properly when the socket read errors occurred.</p>	8.1 SP1
CR102196	<p>In some rare error conditions, there was a memory leak in the <code>NTSocketMuxer</code> when the initiation of an NT asynchronous read failed.</p> <p>This problem was resolved by modifying the <code>NTSocketMuxer</code> to set a "read complete" status correctly after a <code>SocketException</code> occurred.</p>	8.1 SP1
CR102848	<p>The <code>CoreHealthMonitor</code> was holding the <code>ExecuteThreadManager</code> lock while performing significant work, resulting in deadlocks and Administration Console freezes. The code was changed so that <code>CoreHealthMonitor</code> now uses <code>ExecuteThreadRuntimeMBean</code> to get a list of stuck threads and avoids logging from all places in the <code>ExecuteThreadManager</code> if ETM lock is held.</p>	7.0 SP3 8.1 SP1

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR102874 CR103973	By default, network channel tunneling was on, which may have posed a security risk if you had explicitly configured network channels for a WebLogic Server instance.  This problem is resolved by disabling network channel tunneling by default.	8.1 SP1
CR103203	The <code>AcceptBacklog</code> server attribute was erroneously marked as dynamically configurable. This was wrong because the accept backlog for a server socket can only be changed at creation time. WebLogic Server has no dynamic means of recreating server sockets on demand, and so the accept backlog can only be configured statically.  This problem was resolved by ensuring that the accept backlog is only statically configurable.	8.1 SP1
CR103721	Attempts by a Java client to obtain an initial context over t3 using the address of a load balancer as the <code>Provider_URL</code> failed with a connection refused or connection timeout.  A software change now allows a Java client to obtain an initial context over t3, with the <code>Provider_URL</code> set to the IP address of the load balancer. This is only possible when there is not already an established socket connection between the Java client and a server instance in the cluster. Only requests from the Java client that can be addressed to the load balancer are initial context requests.  When using a default channel and t3, it is not necessary to set <code>ExternalDNSName</code> to traverse a NAT firewall. (Initial context requests described above will fail if <code>ExternalDNSName</code> is set.)	7.0 SP3 8.1 SP1
CR103853	If a start-up class or a start-up servlet attempted to set some system properties, a <code>java.util.ConcurrentModificationException</code> was thrown, which would prevent WebLogic Server from starting up.  The problem was resolved by working on a clone of system properties when attempting to log the system properties during startup.	8.1 SP1
CR103967	Child threads of WebLogic Server execute threads did not inherit the correct context classloader. This common problem occurred when a servlet or an EJB created a timer ( <code>java.util.Timer</code> ).  Analysis revealed that WebLogic Server execute threads maintained their own context classloader and child threads of these execute threads did not inherit the context because <code>java.lang.Thread</code> directly assigned the member variable of its own to the child threads.  The problem was solved with a code fix.	6.1 SP6 8.1 SP1

## Resolved Problems for Service Pack 1

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR104030	If an EJB that was not compiled with <code>ejbc</code> or <code>appc</code> , WebLogic Server will automatically compile it at deployment time. The compiler will be picked up from the <code>PATH</code> environment variable. However the <code>commEnv.sh</code> did not export the <code>PATH</code> variable which was causing a wrong compiler to be picked up. This is now resolved.	8.1 SP1
CR104373	<p>When a Java client attempted to invoke the <code>create</code> method on the home stub of a stateless session bean using just the <code>wlclient.jar</code>, a <code>ClassCastException</code> would sometimes occur. This is because the client was unable to retrieve a suitable IIOP stub for the object from the server.</p> <p>This problem was resolved by modifying the stub generator to generate stubs with names that precisely reflect the IIOP Repository IDs used for remote objects.</p>	8.1 SP1
CR107950	<p>Idle RJVMs that lasted for more than 4 minutes were shutdown in the WebLogic Server 8.1 GA version, which was a regression from WebLogic Server 7.0.</p> <p>In WebLogic Server 8.1 SP1, idle RJVMs are not timed out.</p>	8.1 SP1

# Deployment

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR083731	All system-created temporary files are now removed when an auto-deployed application is removed while the Administration Server is down. The corresponding configuration and files are cleaned up the next time the Administration Server is started.	8.1 SP1
CR091897	Deployment failed for a Web application with a short name, such as ab.war or i.war. This error occurred: <pre>java.lang.IllegalArgumentException: Prefix string too short at java.io.File.createTempFile(File.java:1232 at weblogic.j2ee.Component.retrieveComponent(Component.java:296 at weblogic.j2ee.Component.&lt;init&gt;(Component.java:188) at weblogic.j2ee.WebAppComponent.&lt;init&gt;(WebAppComponent.java:45 at weblogic.j2ee.Application.addComponent(Application.java:149) at weblogic.j2ee.J2EEService.addDeployment(J2EEService.java:117)...</pre> The bug was corrected.	6.1 SP5 8.1 SP1
CR099405	Duplicate state transition messages were being logged during deployment operations. This problem was resolved with a code fix.	8.1 SP1
CR101836	A request to deactivate (undeploy) a non-existent JSP using <code>weblogic.Deployer</code> returned a message indicating the operation was successful, rather than returning an error. This problem was resolved by modifying the code so that an error message containing information about how to undeploy a single JSP is returned.	8.1 SP1
CR104230	Javadoc for the <code>weblogic.management.deploy.DeployerRuntime</code> class was modified to remove documentation for extraneous methods.	8.1 SP1

Resolved Problems for Service Pack 1

Change Request Number	Description	Release Fixed
CR104231	<p>Javadoc for the <code>DeployerRuntimeMBean</code> class did not mark deprecated methods. The following methods are now marked as deprecated:</p> <ul style="list-style-type: none"> <li>• <code>activate</code></li> <li>• <code>deactivate</code></li> <li>• <code>remove</code></li> <li>• <code>unprepare</code></li> <li>• <code>lookupActiveTargetsForComponent</code></li> <li>• <code>lookupActiveVirtualHostsFor</code></li> <li>• <code>lookupActiveServers</code></li> </ul>	8.1 SP1
CR104258	<p>If security constraints on a Web application were modified and the Redeploy button for that Web application (in the Administration Console) was clicked, old security constraints were not removed.</p> <p><b>Note:</b> This problem did not occur when the Stop and Deploy buttons were clicked instead of the Redeploy button.</p> <p>This problem was resolved by ensuring that the appropriate methods are called to notify the WebLogic Security Service any time the deployment descriptors are read and security might change.</p>	8.1 SP1
CR108934	<p>Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03-28.01.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03-28.01.jsp</a>.</p>	7.0 SP3 8.1 SP1

# EJB

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR069896	<p>Inside of a CMP deployment descriptor, if a weblogic-rdbms-relation contained two weblogic-relationship-roles with the same relationship-role-name, a NullPointerException could result when the EJB compiler was run.</p> <p>This problem was resolved by adding a compliance check to ensure that the relationship-role-names are unique within a weblogic-rdbms-relation.</p>	8.1 SP1
CR084768	<p>ejbc now logs a warning message when a field group or relationship cache was declared, but not associated with a finder (that is, never used).</p>	8.1 SP1
CR090104	<p>EJBs that specify an isolation level in their descriptor got an exception at runtime. There were two problems. One, WebLogic Server was checking for the wrong vendor workaround; it was checking for <code>supportsTxIsolation</code> instead of <code>supportsTxIsolationUponEnlistment</code>. Two, different database vendors and drivers have different rules/behaviors when (re)setting the isolation level while the transaction is active. For example, when using the WebLogic Server <code>jDriver/XA</code> for Oracle, the isolation level can only be set when creating the transaction branch, but not when joining/resuming the branch.</p> <p>Code changes rectified both the problems.</p>	7.0 SP3 8.1 SP1

Change Request Number	Description	Release Fixed
CR091352	<p>ejbpc code generation was failing for 4-level relationship caching, with the following exception:</p> <pre> Found 4 semantic errors compiling "D:/labs/cluster70/src/com/bea/ps/cmr_orders/ejbcgen/com /bea/ps/cmr_orders/CustomerBean_lq1rpd__WebLogic_CMP_RDB MS.java": 789. if (__WL_bean_orderLines != null) &lt;-----&gt; *** Error: No entity named "__WL_bean_orderLines" was found in this environment. 791. __WL_bean_orderLines.__WL_add_WL_item_field(__WL_eo); &lt;-----&gt; *** Error: "__WL_bean_orderLines" is either a misplaced package name or a non-existent entity. 1310. if (__WL_bean_orderLines != null) { &lt;-----&gt; *** Error: No entity named "__WL_bean_orderLines" was found in this environment. 1312. __WL_bean_orderLines.__WL_add_WL_item_field(__WL_eo); &lt;-----&gt; *** Error: "__WL_bean_orderLines" is either a misplaced package name or a non-existent entity. </pre> <p>Analysis revealed that RDBMSCodeGenerator.java was trimming an extra character when creating the prevCmrFieldName string, when caching was more than 3 levels.</p> <p>A fix to RDBMSCodeGenerator.java solved the problem.</p>	7.0 SP3 8.1 SP1
CR092202	<p>WebLogic Server did not check that an Oracle sequence existed when a bean was deployed.</p> <p>This problem was resolved by adding the following behaviors:</p> <ul style="list-style-type: none"> <li>• The create/alter feature works in non-production mode only.</li> <li>• The container never drops a sequence.</li> <li>• If the container creates a sequence, it leaves an indication that the sequence has been created. (The indicator has the format: &lt;sequence-name&gt;_created_by_webLogic).</li> <li>• The container may only alter a sequence that has an indicator in the database.</li> <li>• When deploying in development mode, the container looks for &lt;sequence-name&gt; and if found, uses that sequence. Otherwise, the container looks for &lt;sequence-name&gt;_created_by_weblogic indicator.</li> </ul>	8.1 SP1

Change Request Number	Description	Release Fixed
CR094629 CR098386	<p>With a CMR one-to-one (created in the same transaction), a database <code>SELECT</code> is issued to ensure the database consistence. This database <code>SELECT</code> can be avoided in some cases, if the proper constraints are set on the database.</p> <p>This database <code>SELECT</code> is now avoided by modifying the code so that <code>__WL_CMR_isLoaded_</code> is set to true. This causes the database round-trip to be skipped when the bean state is <code>STATE_EJB_POSTCREATE</code>. Because this resolution causes the database check to be skipped, customers must be sure to create the database with the proper constraints.</p> <p>This resolution is limited to the case where a bean that has a 1 to 1 relationship with the current bean is being created in one of the bean's <code>ejbPostCreate</code> methods, and the CMR field in a bean's create method is implicitly set using a CMP field that is mapped to the same CMR field. For example, a parent's primary key is embedded as part of child's CMR field. If the CMR field setter is used, the container will still do the database <code>SELECT</code>.</p>	7.0 SP3 8.1 SP1
CR094871	<p><code>O_ITEM</code> table (<code>ItemEnt</code>) caused a deadlock in <code>SPECjAppServer</code>. This deadlock only occurred with deployment descriptors that would be used for a single-server configuration, not for a multiple-server configuration.</p> <p>This deadlock was caused by different orderings of the primary keys in two different transactions for the SQL statement that was used for read verification (for the <code>verify-rows</code> feature).</p> <p>This problem was resolved with a code fix.</p>	8.1 SP1
CR095141	<p>The <code>CoreHealthMonitor</code> was holding the <code>ExecuteThreadManager</code> lock while performing significant work, resulting in deadlocks and Administration Console freezes. The code was changed so that <code>CoreHealthMontor</code> now uses <code>ExecuteThreadRuntimeMBean</code> to get a list of stuck threads and avoids logging from all places in the <code>ExecuteThreadManager</code> if <code>ETM</code> lock is held.</p>	7.0 SP3 8.1 SP1

Change Request Number	Description	Release Fixed
CR095173	<p>The <code>idle-timeout-seconds</code> element determines how long the EJB container waits before passivating stateful session beans, that is, removing them from cache and writing them to disk. The EJB container also used to use this element to determine how long to wait before removing passivated EJBs from the disk. However, some users wanted stateful session beans to remain on disk longer than <code>idle-timeout-seconds</code>. In other words, they want to specify how long stateful session beans stay idle in the cache and how long they stay idle on disk using two different elements.</p> <p>WebLogic Server 7.0 Service Pack 3 introduced the new element <code>session-timeout-seconds</code>, which specifies how long the EJB container waits before removing an idle stateful session bean from disk. This element is also available in WebLogic Server 8.1 Service Pack 1.</p>	7.0 SP3 8.1 SP1
CR096800	<p>When WebLogic Server 6.1 throws an EJB exception to a WebLogic Server 8.1 invoker, the invoker received a serialization error. This is because WebLogic Server 6.1 shipped with the wrong serial version unique identifier (SVUID) for <code>EJBException</code>.</p> <p>WebLogic Server 8.1 SP1 identifies when it is communicating with WebLogic Server 6.x and accepts and sends the incorrect SVUID over such connections. The earlier resolution attempt along these lines that shipped with 8.1 was flawed.</p>	7.0 SP3 8.1 SP1
CR097155	<p>There were several issues with the <code>default-dbms-tables-ddl</code> feature:</p> <ul style="list-style-type: none"> <li>• It did not create the <code>ddl</code> file when the <code>EJBCompiler</code> was run</li> <li>• It was not clear where the <code>ddl</code> file was placed once it was created</li> <li>• If the <code>ddl</code> file already existed, it was modified</li> </ul> <p>These problems were resolved by modifying the code so that the <code>ddl</code> file is now created at compile time. A message indicates the location of the <code>ddl</code> file when it is written, and if the <code>ddl</code> file already exists, it is overwritten.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR098343	<p>JDBC connections were not returned to the pool when a transaction was distributed over two server instances. The JDBC attribute <code>KeepXAConnTillTxComplete</code> was set to <code>true</code>. This sequence of events led to the problem:</p> <ol style="list-style-type: none"> <li>1. Start a user transaction on one server instance 1.</li> <li>2. Put a JMS message into a persistent queue. The JMS server and queues are on server instance 2.</li> <li>3. Update an EJB on server instance 2.</li> <li>4. Commit the transaction.</li> </ol> <p>No error messages were generated. The updates were performed and the message was placed in the JMS queue. However, some connections remained 'in use' in the EJB connection pool. After repeating the sequence of steps several times, the connection pool ran out of connections.</p> <p>The problem did not occur if the actions were executed on a single server instance, or if the JMS message was placed in the queue outside the transaction scope.</p> <p>The problem was solved with a code change to release the connection in the <code>afterCompletion</code> callback.</p>	6.1 SP6 8.1 SP1
CR098548	<p>A compliance warning has been added indicating that corruption of a Blob Clob may occur if a bean is using optimistic concurrency, has any fields that are mapped to Blob/Clob DBMS columns, and no timestamp or version column has been specified. This warning also recommends the addition of a timestamp or version column.</p>	8.1 SP1
CR098798	<p>When the EJB compiler was unable find a method specified in a transaction attribute setting, isolation level setting, method permission setting or an idempotent method setting, the error message produced did not contain the particular EJB on which the method could not be found.</p> <p>The error message was augmented to contain this information.</p>	8.1 SP1
CR100010	<p>An expired <code>license.bea</code> file resulted in a <code>NullPointerException</code> when attempting to deploy an EJB, rather than an exception indicating a problem with the license.</p> <p>This problem was resolved by modifying the code so that no deployments can be attempted after failing the license check, and a message indicates a problem with the license.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR100719	<p><code>appc</code> previously did not take <code>alt-dd</code> descriptors into account when invoking the underlying module compilers. The module compilers now use <code>alt-dd</code> descriptors if they are defined in the <code>application.xml</code> descriptor of the EAR. Command-line options were also added to <code>appc</code> to support application-level <code>alt-dds</code>.</p>	8.1 SP1
CR100822	<p>The behavior of clustered message-driven beans has changed to optimize performance. Previously, only standard cluster load-balancing algorithms were used when an MDB connected to JMS, so that JMS connections were balanced across the cluster.</p> <p>This has been augmented so that, when possible, an MDB steers its JMS connection toward the same JVM that is hosting the messages from which the bean receives connections. This reduces the number of hops made by the MDB when it is accessing the resource.</p>	7.0 SP3 8.1 SP1
CR100890	<p><code>ejbc</code> now recognizes EJB QL identifiers with acceptable UNICODE names. <code>ejbc</code> warns if any non-standard characters are used in an identifier but allows them to be used. Continued use of non-standard characters in an identifier may result in non-portable applications across J2EE containers.</p>	8.1 SP1
CR101248	<p>CMP beans were leaking JDBC connection pools. The problem was occurring when <code>ejbFindByPrimaryKey()</code> called <code>releaseResources()</code>.</p> <p>A change was made to the <code>RDBMSCodeGenerator</code> so that, if <code>ResumeTransaction</code> fails, WebLogic Server releases resources to avoid a potential JDBC connection leak. This resolved the problem.</p>	7.0 SP3 8.1 SP1
CR101446	<p>In some cases, the EJB compiler was generating an interface that illegally declared methods as <code>final</code>.</p> <p>This problem was resolved by modifying the code generator to not declare interface methods with a <code>final</code> modifier.</p>	8.1 SP1
CR102308	<p>The Administration Console reported incorrect values for waiters for entity beans.</p> <p>The problem was solved by adding a <code>waiterCurrentCount</code> attribute that is incremented when a client starts waiting for a lock, and decremented when the lock acquired or the client times out.</p>	6.1 SP6 8.1 SP1

Change Request Number	Description	Release Fixed
CR102565	<p>An incorrect <code>javac</code> CLASSPATH may have caused compiler errors in <code>appc</code>. This problem was particularly likely if one version of an EJB class existed in the system CLASSPATH and a different version existed in the JAR file. This problem may also have affected compilers other than <code>appc</code>.</p> <p>This problem occurred because of an incorrect ordering of the JAR file in the system CLASSPATH for the <code>appc</code> invocation.</p> <p>This problem was resolved by modifying the code so that the CLASSPATH passed to <code>javac</code> lists the system CLASSPATH before the PATH to the JAR file being compiled.</p>	8.1 SP1
CR103108	<p>EJB runtime counts that were of type <code>long</code> were not being synchronized, resulting in corrupt and/or incorrect count results.</p> <p>This problem was resolved with a code fix.</p>	8.1 SP1
CR103226	<p>Relationship caching for CMP is now supported for SQLServer 2000.</p> <p>Using nested relationship caching in SPECjAppServer 2002 using SQL Server 2000 used to cause the following SQL Server 2002 error:</p> <pre>java.sql.SQLException: [BEA][SQLServer JDBC Driver][SQLServer]Query contains an outer-join request that is not permitted.</pre> <p>This problem was resolved by modifying the outer join syntax to handle multiple database types, and by adding a <code>&lt;database-type&gt;</code> tag to the <code>weblogic-rdbms-jar</code> file. This tag indicates the database type of the underlying DBMS. For SQL Server 2000, specify the <code>&lt;database-type&gt;</code> as <code>SQLServer2000</code>.</p> <p><b>Note:</b> SQL Server 7 should be specified in the <code>&lt;database-type&gt;</code> tag as <code>SQLServer</code>.</p>	8.1 SP1
CR103393	<p>A bug in the CMP code generator sometimes resulted in generated code that would not compile when a CMP field was also a CMR field. In particular, an array was being accessed with an invalid index (the index was <code>null</code>). This problem was rare and only showed up when the EJB compiler was invoked.</p> <p>This problem was resolved by ensuring that the code generator always used a valid array index.</p>	8.1 SP1
CR103341	<p>New convenience functionality for CMP beans using Oracle sequences was added. The tag <code>&lt;create-default-dbms-tables&gt;</code> will now operate on Oracle sequences on systems that are being used in development mode.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR104046	<p>With <code>delay-database-insert-until</code> set to <code>Commit</code>, the Optimistic concurrency version column was not being initialized when a "multi-table" bean was created. The code was changed so that <code>RDEMSCodeGenerator.perhapsAssignOptimisticField()</code> now returns generated code that sets the version column for all the tables that the CMP bean is mapped to.</p>	
CR104256	<p>CMP Bean creation failed if any of the primary key field values were Java primitives and if any of these primitive key field values were set to a value equal to the default value for that field type. For example, if a bean had a primary key field of type Java primitive <code>'int'</code> and the value of the primary key field was <code>'0'</code>, then CMP bean creation would fail.</p> <p>The CMP container now allows the setting of Java primitive default values for primary key fields to be recognized and for EJB creation to proceed. This fixes a bug in the container. No other side effects are expected.</p>	8.1 SP1
CR104610	<p>By default, <code>appc</code> would fork a new process for compiling. In some cases (large EARs, for example), this process required customers to allocate a lot of memory to the JVM heap to successfully compile, and sometimes a <code>java.lang.OutOfMemoryError</code> would occur anyway.</p> <p>This problem was resolved by adding an explicit <code>-J</code> option to the <code>wlappc</code> ant task's <code>runtimeFlags</code> option. Customers can now pass a memory size to the new process, such as <code>runtimeFlags="-J-ms64m -J-mx256m"</code>. Additionally, the default for <code>appc</code> was changed to "inline compilation". Therefore, <code>appc</code> will no longer fork a new process for compiling by default.</p> <p>For example, in WebLogic Server 8.1 GA:</p> <pre data-bbox="266 1216 1022 1350">&lt;wlappc forceGeneration="true" compiler="javac" runtimeFlags="-ms64m" lineNumbers="true" verbose="true" debug="true" keepgenerated="true" source="\${dist-dir}/ejb_\${name}.jar"&gt; &lt;classpath refid="ejbc-classpath" /&gt; &lt;/wlappc&gt;</pre> <p>In 8.1 SP1:</p> <pre data-bbox="266 1407 928 1541">&lt;wlappc forceGeneration="true" compiler="javac" runtimeFlags="-J-ms64m" lineNumbers="true" verbose="true" debug="true" keepgenerated="true" source="\${dist-dir}/ejb_\${name}.jar"&gt; &lt;classpath refid="ejbc-classpath" /&gt; &lt;/wlappc&gt;</pre>	8.1 SP1

Change Request Number	Description	Release Fixed
CR104979	<p>This change affects the following EJB callback methods on the following bean types:</p> <ul style="list-style-type: none"> <li>• <code>ejbCreate</code>, <code>ejbRemove</code>—for stateless session beans and message driven beans.</li> <li>• <code>ejbPassivate</code>—for stateful session beans when <code>ejbPassivate</code> is called due to a cache timeout.</li> </ul> <p>The security principal that is active when these methods are called will be anonymous except if any of the following are done:</p> <ol style="list-style-type: none"> <li>1. For those beans which have a <code>run-as</code> role specified, a <code>run-as</code> principal has been designated according to the rules for designating a <code>run-as</code> principal. In these cases the <code>run-as</code> principal will be active.</li> <li>2. If the bean is new for this service pack: <code>&lt;create-as-principal-name&gt;</code> or <code>&lt;remove-as-principal-name&gt;</code> or <code>&lt;passivate-as-principal-name&gt;</code> are specified for the corresponding EJB callback methods, then the specified principal is active during the EJB callback method.</li> </ol> <p>Note that principals specified using this method will take precedence over the more general <code>run-as</code> role principal assignment that is specified using (1) above.</p> <p>If any operations performed as part of these EJB callbacks require more permissions than anonymous would have, then an appropriate principal must be set using one of the methods (1) or (2) above.</p>	8.1 SP1
CR107455	<p>When read-only beans were used, <code>OutOfMemoryErrors</code> occurred.</p> <p>This problem occurred because multiple duplicate entries were being created for the same bean in the <code>lastLoadMap</code>. The duplicate entries were being created by an unnecessary call to <code>initLastLoad()</code> during <code>getBeanFromRS()</code>, even though <code>lastLoadMap</code> was already initialized by the call to <code>cache.get()</code>.</p> <p>This problem was resolved by removing the unnecessary call to <code>initLastLoad()</code>.</p>	8.1 SP1

## Installation

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR098373	<p>The WebLogic Server installer did not provide users with the ability to exit the installer when it was determined that the available space was not adequate for product installation. Instead, the installer only provided users with an option to specify another directory.</p> <p>This problem was resolved by providing an option for users to exit the installer when it was determined that the available space was not adequate for product installation.</p>	8.1 SP1
CR101452	<p>The name of the JDK directory was hardcoded in the <code>config.cmd</code> and <code>config.sh</code> scripts.</p> <p>This problem was resolved by replacing the hardcoded JDK directories to <code>%JAVA_HOME%\bin\java</code> and <code>%JAVA_HOME%\bin\javaw</code>.</p>	8.1 SP1

# Internationalization

Change Request Number	Description	Release Fixed
CR099768	<p>If the catalogs listed on the command line do not have any translations, <code>l10ngen</code> exits abruptly but does not provide an error message indicating the reason.</p> <p>This problem was resolved by modifying the code so that if <i>none</i> of the catalogs have translations, a warning is issued. If <i>any</i> catalogs have translations, then the translations are processed, but no warnings are issued for those that do not have translations.</p>	8.1 SP1
CR102646	<p>When attempting to convert I18N message files to WebLogic Server message catalogs, the use of the <code>Loggable</code> class caused an error like the following to be written to the server log (rather than the appropriate message):</p> <pre>&lt;Apr 1, 2003 1:31:35 PM PST&gt; &lt;Error&gt; &lt;Unknown&gt; &lt;000000&gt; &lt;Unable to access undefined message, id=900017&gt;</pre> <p>This problem occurred because the <code>Loggable</code> class attempted to look up the message catalog class name based on the message ID, which would not work properly if there were issues with the classloader.</p> <p>This problem was resolved by modifying the code so that the message catalog class name is now passed into the constructor of the <code>Loggable</code> class.</p>	8.1 SP1
CR107052	<p>Internationalization characters in the <code>InternalRequestDispatcher</code>'s response were garbled because the <code>String</code> content was created without the charset.</p> <p>This problem was resolved by creating the <code>String</code> with the appropriate charset, thus ensuring that the <code>String</code> appears in the correct encoding. The change should cause no side-effects.</p>	8.1 SP1

## J2EE

Change Request Number	Description	Release Fixed
CR091939	<p>In <code>weblogic.Deployer</code>, an application would always use the deployment descriptor with which it was originally deployed. In other words, you could not change the deployment descriptor when the application was redeployed using the <code>-redeploy</code> flag. (You could only change the deployment descriptor if you first used the <code>-undeploy</code> flag.)</p> <p>This problem was resolved by modifying the code to allow the switching of deployment descriptors at update time. The behavior of the <code>-redeploy</code> flag now works the same way as an <code>-undeploy</code> followed by a <code>-redeploy</code>.</p>	8.1 SP1
CR096215	<p>Any war file containing a <code>WEB-INF/web-services.xml</code> deployment descriptor will now have its associated Web Services compliance checked when compiled by <code>appc</code>. This additional checking was added to help identify application errors more quickly during development.</p>	8.1 SP1
CR098671	<p>A split-directory EAR could not be deployed with a module-level <code>alt-dd</code> deployment descriptor.</p> <p>This problem was resolved by a code fix. For more information about using module-level <code>alt-dd</code> deployment descriptor, see the “Module” table under <a href="#">“Application Deployment Descriptor Elements”</a> in <i>Developing WebLogic Server Applications</i>.</p>	8.1 SP1
CR101002	<p>When an EAR was deployed security roles defined in the <code>application.xml</code> file but with no corresponding <code>security-role-assignment</code> entries in <code>weblogic-application.xml</code>, these security roles were not visible from the WebLogic Server Administration Console when right-clicking the deployed enterprise application and selecting “Define Scoped Role...”.</p> <p>This problem was resolved by modifying the code so that all security roles defined in <code>application.xml</code> are viewable in the Administration Console (even those without assignments in <code>weblogic-application.xml</code>).</p>	8.1 SP1
CR102293	<p><code>wlcompile</code> supports a nested <code>javac</code> element that you can use to specify flags on <code>javac</code>. For example, you can specify deprecation warnings on <code>javac</code> as follows:</p> <pre>&lt;wlcompile srcdir="{srcdir}" destdir="{destdir}"&gt;   &lt;javac deprecation="true" /&gt; &lt;/wlcompile&gt;</pre>	8.1 SP1

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR102306	<p>The <code>wldeploy</code> command did not support application-level <code>alt-dd</code> deployer arguments. Attempts to use such arguments would result in a build failure.</p> <p>This problem was resolved by adding support to <code>wldeploy</code> for application-level <code>alt-dd</code> deployer arguments. This means that split-directory EARs can now be deployed with application-level <code>alt-dds</code>.</p>	8.1 SP1
CR105317	<p>When specifying preprocessor names on the command line as shown below, any spaces before or after the classname were interpreted as part of the classname, and therefore, an otherwise legitimate class would not be found.</p> <pre>-Dweblogic.classloader.preprocessor="com.dirig.preprocessor.DirigBEAClassProcessor, ppclass.MyPP1, ppclass.MyPP2 "</pre> <p>This problem was resolved by ensuring that any such spaces are not interpreted as part of the classname.</p>	8.1 SP1

## JDBC

<b>Change Request Number</b>	<b>Description</b>	<b>Released Fixed</b>
CR090255	<p>Version 9.2.0.2 of the Oracle Thin driver, which shipped with the product, produced a <code>NullPointerException</code> when the <code>addBatch</code> and <code>setNull</code> methods required data conversions.</p> <p>This problem was resolved by updating the Oracle Thin driver to version 9.0.2.3.</p>	8.1 SP1
CR090365 CR090366	<p>Version 9.2 of the Oracle Thin driver, which shipped with the product, did not allow a BLOB and a long raw to coexist in the same table. The following exception would be encountered:</p> <pre>java.sql.SQLException: Protocol violation</pre> <p>This problem was resolved by updating the Oracle Thin driver to version 9.0.2.3.</p>	8.1 SP1

Change Request Number	Description	Released Fixed
CR092453	<p>In WebLogic Server 6.1 SP04, with Oracle Thin XA with the CLASSES12.zip from Oracle 9.2, a stateless session bean calling EJB caused XAER_PROTO after "Configuration Changes saved to the repository" message appeared.</p> <pre>START SLEEP 2: After updating the value to 1... DONE SLEEP 2: After updating the value to 1... START SLEEP 2: After updating the value to 2... DONE SLEEP 2: After updating the value to 2... START SLEEP 2: After updating the value to 3... DONE SLEEP 2: After updating the value to 3... START SLEEP 2: After updating the value to 4... DONE SLEEP 2: After updating the value to 4... START SLEEP 2: After updating the value to 5... DONE SLEEP 2: After updating the value to 5... Current value is 5</pre> <p>&lt;Dec 6, 2002 10:26:59 PM MST&gt; &lt;Info&gt; &lt;Management&gt;  &lt;Configuration changes for domain saved to the repository.&gt; SQLException -- XA error: XAER_PROTO :  Routine was invoked in an improper context start() failed  on resource 'OracleXA' null Current value is 0</p> <p>The problem caused by a known problem in Oracle client 9.2.0.[01], that is solved in 9.2.0.2. WebLogic Server implemented a code change to work around the 9.2.0.[01] issue.</p>	6.1 SP5 8.1 SP1
CR094729	<p>The JDBCConnectionPoolRuntimeMBean.getStatementProfiles method did not filter results by poolName for the MBean instance it was called against. If you called the getStatementProfiles method, the results included statements for all connection pools for which tracing had been activated. This was incorrect because the JDBCConnectionPoolRuntimeMBean instance is specific to a single connection pool.</p> <p>This problem was resolved by modifying the JDBCConnectionPoolRuntimeMBean.getStatementProfiles method to correctly filter results by poolName for the MBean instance it is called against.</p>	7.0 SP3 8.1 SP1

Change Request Number	Description	Released Fixed
CR099872	<p>The message for the Exception during commit of transaction stack trace exception contained the connection pool name, but not the data source name.</p> <pre>&lt;Exception during commit of transaction Xid=39:74a54046e2c2bb30(5962554),Status=Rolled back. [Reason=ja vax.transaction.xa.XAException: JDBC driver does not support XA, hence cannot be a participant in two-phase commit. To force this participation, set the EnableTwoPhaseCommit property on the corresponding JDBCTxDataSource property, to true. Pool = CatPhase1] ...</pre> <p>The stack trace content was enhanced to include the name of the data source.</p>	6.1 SP6 8.1 SP1
CR100877	<p>The JTS driver attempted to create remote connections in single-server SpecJAppServer runs.</p> <p>This problem occurred only if a connection to the database for the transaction had already been opened, and the getConnection method was called again when the transaction timed out and was rolled back. In addition, this problem:</p> <ul style="list-style-type: none"> <li>• Occurred infrequently.</li> <li>• Resulted in no impact to the behavior of the application.</li> <li>• Occurred as a result of a rollback race condition.</li> <li>• Could have negatively impacted performance.</li> </ul> <p>This problem was resolved by modifying the code so that the transaction status is checked before a remote connection is made.</p>	8.1 SP1
CR101419	<p>Fail-over for DataSources and TxDataSources in a cluster was not working.</p> <p>This problem occurred because the DataSources and TxDataSources were not built as clusterable, and was resolved by building them as clusterable.</p>	8.1 SP1

Change Request Number	Description	Released Fixed
CR101709	<p>DB vendors (such as Oracle) execute SQL DDL operations within a new local transaction. However, if the <code>AutoCommit</code> flag for the connection was set to false (as is the case when a connection is obtained from a WebLogic Server <code>TxDataSource</code>), the local transaction was not committed, and remained active on the connection.</p> <p>When running with the Oracle 9.2.0 Thin driver, subsequent XA operations performed on this connection failed with an <code>XAER_PROTO</code>, due to the currently active local transaction. This is because starting with this version, Oracle Thin driver started checking for active local transactions when asked to perform XA operations on connection objects.</p> <p>WebLogic Server can now circumvent this problem by rolling back any active local transactions on a connection when it is being returned back to the WebLogic Server JDBC connection pool. This defensive behavior can be enabled by setting the new WebLogic Server JDBC connection pool attribute <code>RollbackLocalTxUponConnClose</code> to true.</p>	8.1 SP1
CR101822	<p>Setting the isolation level to <code>Connection.TRANSACTION_READ_COMMITTED</code> for Oracle Thin Driver failed, with an exception indicating that the Isolation Level "2" could not be mapped.</p> <p>This isolation level is now properly recognized and passed to the Oracle driver.</p>	8.1 SP1
CR102364	<p>Clicking the Test Pool button on the Testing tab in the Administration Console after creating an Informix JDBC connection pool might have resulted in the following errors:</p> <pre data-bbox="266 1159 1032 1359"> &lt;Error&gt; &lt;JDBC&gt; &lt;BEA-001112&gt; &lt;Test "select count(*) from ITEMS" set up for pool "informixPool" failed with exception: "java.sql.SQLException: Database not selected yet."&gt; &lt;Error&gt; &lt;JDBC&gt; &lt;BEA-001111&gt; &lt;Unable to verify the test "select count(*) from ITEMS" set up for pool "informixPool". Connections will not be tested.&gt; </pre> <p><b>Note:</b> This problem only occurred for some Informix configurations.</p> <p>This problem was resolved by modifying the code so that the configuration works properly for Informix JDBC connection pools.</p>	8.1 SP1

Change Request Number	Description	Released Fixed
CR102698	<p>Attempts to create a connection pool with <code>weblogic.Admin CREATE_POOL</code> resulted in the following exception:</p> <pre>./wlg-create-pool.sh No permission to create ConnectionPool Start server side stack trace: weblogic.common.ResourceException: No permission to create ConnectionPool at weblogic.jdbc.common.internal.JDBCService.createPool(Lj ava.util.Properties;Lweblogic.secur ity.acl.internal.AuthenticatedSubject;)V(Unknown Source) at weblogic.jdbc.common.internal.ConnectionPool.createPool (Ljava.util.Properties;Lweblogic.se curity.acl.internal.AuthenticatedSubject;)V(Unknown Source) at ...</pre> <p>Research revealed that WebLogic Server was requiring <code>aclName</code>—which is deprecated—as an attribute. A code fix resolved the problem.</p>	7.0 SP3 8.1 SP1
CR103321	<p>The <code>getConnectionsTotalCount()</code> method of <code>JDBCConnectionPoolRuntimeMBean</code> did not behave as expected. Instead of returning the total number of JDBC connections in the pool since instantiation, it returned the maximum number of connections since instantiation.</p> <p>The problem was resolved with a code fix in the method.</p>	8.1 SP1
CR104103	<p>RowSets failed to serialize a CLOB column to XML, resulting in an exception similar to the following:</p> <pre>java.io.IOException: No XML Schema type mapping found for JDBC type: 2005</pre> <p>This problem was resolved by modifying the code so that a CLOB is appropriately mapped to an XML schema type.</p>	8.1 SP1
CR104522	<p>The <code>CachedRow.containsKey(Object)</code> method always returned an inverted value.</p> <p>This problem was resolved with a code fix.</p>	8.1 SP1
CR104523	<p><code>DatabaseMetaData.getDriverVersion()</code> was returning an outdated version string. A code fix resolved the problem.</p>	7.0 SP3 8.1 SP1

Change Request Number	Description	Released Fixed
CR107609	<p>If a RowSet had to look up the DataSource itself (instead of it being set from outside), an application was forced to set the username and password.</p> <p>This was due to a problem in the cleanup processing for RowSets. The security check that occurs when JNDI credentials are used reduced performance by about 40% in some cases.</p> <p>This problem was resolved by fixing the problem in the cleanup processing.</p>	8.1 SP1
CR108006	<p>When multiple transactional (JTS) JDBC connections were used in a remote application during a single transaction, a <code>NullPointerException</code> could be seen on the remote client if any of the connections were closed and any other connections were used after that.</p> <p>This problem occurred because, while all RMI connections in the same JTS transaction must actually represent the same underlying JTS connection, a mistaken optimization was added, which supplied the <i>same</i> RMI connection for multiple connection requests within a transaction. Closing any one of these would also close what the application thought was any other, supposedly independent one.</p> <p>The solution was to remove the optimization, and to always return distinct RMI connections to separate remote connection requests within a given JTS transaction.</p>	8.1 SP1

## JMS

Change Request Number	Description	Release Fixed
CR097038	<p>When a source JMS bridge destination was not running in the same WebLogic Server 8.1 domain as the Message Bridge, then a trusted security relationship must have been established between the domains. Otherwise, the Messaging Bridge must have been configured to work in synchronous mode (Asynchronous Mode Enabled = No).</p> <p>This problem has been resolved so that non-trusted domains also work when using asynchronous mode communications.</p>	7.0 SP3
CR102656		8.1 SP1

Change Request Number	Description	Release Fixed
CR098280	<p>The session creation method behavior has been changed when using a <code>XAQueueConnection</code> or <code>XATopicConnection</code> object to create a non-XA Session. Prior to this change, the behavior was as follows:</p> <ul style="list-style-type: none"> <li>• <code>XAQueueConnection.createQueueSession</code> creates a <code>XAQueueSession</code></li> <li>• <code>XATopicConnection.createTopicSession</code> creates a <code>XATopicSession</code></li> </ul> <p>In both cases, the user-specified acknowledge mode and transacted flag were ignored and replaced with the <code>AUTO_ACKNOWLEDGE</code> mode and a transacted setting of <i>false</i>.</p> <p>Whereas, the new behavior causes the <code>XAQueueConnection.createQueueSession</code> and <code>XATopicConnection.createTopicSession</code> methods to behave <i>exactly</i> the same as the corresponding methods on <code>QueueConnection</code> and <code>TopicConnection</code>, as follows:</p> <ul style="list-style-type: none"> <li>• <code>QueueConnection.createQueueSession</code> creates a <code>QueueSession</code></li> <li>• <code>TopicConnection.createTopicSession</code> creates a <code>TopicSession</code></li> <li>• <code>XAQueueConnection.createQueueSession</code> creates a <code>QueueSession</code></li> <li>• <code>XATopicConnection.createTopicSession</code> creates a <code>TopicSession</code></li> </ul> <p>Furthermore, the user-specified acknowledge mode and transacted flag settings will be honored for each of these four methods.</p> <p>The four connection methods listed above behave differently if the <b>XAServerEnabled</b> flag is enabled on the connection factory. If this flag is enabled, then all four methods will create an <code>XAQueueSession</code> (or <code>XATopicSession</code>) session if invoked on the server, and a non-XA <code>QueueSession</code> or <code>TopicSession</code> session if invoked on the client. The resulting session will honor the user-specified acknowledge mode, but it will ignore the transacted flag since the resulting session supports XA.</p> <p>Note that in previous versions of WebLogic Server, connection objects that were created from a connection factory with the <b>XAConnectionFactoryEnabled</b> flag enabled behaved as if they were <code>XAQueueConnection</code> or <code>XATopicConnection</code> objects. With the new behavioral change, this behavior is now invisible unless you explicitly cast the connection factory to <code>XAQueueConnectionFactory</code> or <code>XATopicConnectionFactory</code> and called one of the <code>createXA</code> methods.</p> <p>Prior to this change, if you set the <b>XAConnectionFactoryEnabled</b> flag on your connection factory, you would have noticed different behavior from the <code>createQueueSession</code> and <code>createTopicSession</code>, even if you did not cast the connection factory to one of the XA connection factory classes.</p>	7.0 SP3 8.1 SP1

Change Request Number	Description	Release Fixed
CR098975	<p>WebLogic Server sometimes failed to boot after you undeployed a distributed destination. This occurred when you deleted the distributed members and the distributed destination itself, but did not delete the underlying physical destinations. The following sequence of Administration Console actions caused this problem:</p> <ol style="list-style-type: none"> <li>1. In the Configuration -&gt; Members tab of a distributed destination, click the trash can icon to remove a member.</li> <li>2. When prompted, do not delete the member's underlying physical destination.</li> <li>3. Click Remove to delete the member.</li> <li>4. If necessary, continue deleting all the members of the distributed destination.</li> <li>5. Delete the distributed destination.</li> </ol>	8.1 SP1
	<p>If you had shut down the server at this point, you would not be able to reboot it.</p> <p>The problem was resolved by creating a JMS template for distributed destinations at the domain level. This prevents the template from being deleted when a user deletes a distributed destination member but does not delete the underlying physical destination represented by the member. This way, when WebLogic Server boots, the JMS template that the physical destination is referencing still exists so the boot does not fail.</p>	
CR100663	<p>A new debug flag "DebugJMSMessagePath" was added to trace the JMS message in the server and more debugging messages were added. When the new DebugJMSMessagePath flag is turned on, the JMS message inside the server is tracked and a debug message is logged informing the message's path along with the JMSMessageID, if available.</p> <p>This flag can be turned in two ways as shown below:</p> <ul style="list-style-type: none"> <li>• In the config.xml: <pre data-bbox="270 1211 637 1315">&lt;ServerDebug   Server="name of the server"   DebugJMSMessagePath="true" /&gt;</pre> </li> <li>• In the server startup command-line as: <pre data-bbox="270 1367 834 1390">-Dweblogic.Debug.DebugJMSMessagePath=true</pre> </li> </ul>	8.1 SP1
CR101076	<p>The messaging bridge was enhanced to log all JMS header attributes when -Dweblogic.Debug.DebugMessagingBridgeRuntime is set to true.</p>	8.1 SP1

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR101298	<p>If a long idle JMS JDBC connection was marked as “bad” (when for example, if a firewall time-to-live expires) JMS attempted to use the now bad connection, and would fail. The failure would either be immediate or would take upwards of 8-10 minutes to return, depending on tuning parameters.</p> <p>This problem was resolved by modifying the code so that if JMS is idle, the database is pinged every 5 minutes to maintain the connection.</p>	7.0 SP3 8.1 SP1
CR101585	<p>A JMS Client Session that was never committed or acknowledged would result in a memory leak. This problem was resolved by properly handling rollback and recovery when unacknowledged messages exist.</p>	8.1 SP1
CR101594	<p>When the WebLogic Server licence file did not include a license for JMS and a JMS Server was configured and targeted to a WebLogic Server instance, the server booted without the JMS Server, but did not indicate the reason for the JMS Server’s failure to start in the server log file.</p> <p>This problem was resolved by logging an error in this case.</p>	8.1 SP1
CR101804	<p>When undeploying a JMS server by setting the JMS server’s target to “none” in the Administration Console, redeploying the JMS server by setting its target back to the original WebLogic server instance resulted in a <code>NullPointerException</code>, and the redeploy failed. This problem was more likely to have occurred when JMS templates were used.</p> <p>This problem was resolved.</p>	8.1 SP1
CR102234	<p>ASSERTION FAILED messages were printed to the bridge server log when a server was killed.</p> <p>This problem occurred because the bridge adapter code closed both the source connection and the target connection, when one of these connections may already have been destroyed and removed from the pool.</p> <p>This problem was resolved by modifying the bridge adapter code so that it does not attempt to close a bad connection.</p>	8.1 SP1
CR102284	<p>The performance of the JMSHelper extension methods was improved.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR102649	<p>Errors [J2EE:160054] and [J2EE:160062] frequently occurred in the case that an application was using JMS connection pooling, where the session was opened in <code>ejbCreate()</code> and closed in <code>ejbRemove()</code>.</p> <p>This problem occurred because the EJB was returned to the pool before the transaction was complete, and the JMS session was still associated with the old transaction. This problem was resolved.</p>	8.1 SP1
CR102656	<p>Using a messaging bridge for asynchronous messages between non-trusted domains resulted in security error messages.</p> <p>The code was changed to handle message passing between the domains with allowable security credentials in this case.</p>	8.1 SP1
CR102749	<p>JMS JDBC stores may have deleted durable subscriptions over time. If a durable subscription record was accidentally deleted, its associated messages were also deleted on the next reboot of WebLogic Server.</p> <p>This problem was fixed by keeping track of handles for durable subscriptions so that wrapping handle values would not re-use an existing handle.</p>	8.1 SP1
CR103375	<p>Shutting down a server resulted in the following exception:</p> <pre>weblogic.jms.common.JMSEException: Failed to remove temporary destination because JMSServer is shutdown or suspended ...</pre> <p>This problem occurred because attempts to remove temporary destinations are made while the server is in the shutting down state, and therefore these attempts cannot be completed.</p> <p>This problem was resolved by modifying the code so that a <code>JMSEException</code> is not thrown when attempts to remove a temporary destination are made during a server shutdown, and the temporary destinations are successfully removed.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR105337	<p>Under load conditions, a test JMS application was encountering the following exception:</p> <pre> java.lang.OutOfMemoryError weblogic.jms.common.TransactionRollobackException: at weblogic.jms.backend.BEConsumer.expireTimeout(BEConsumer.java:1620) at weblogic.jms.backend.BEXATranEntryBlockingConsumer.startRollback(BEXATranEntryBlockingConsumer.java:72) at weblogic.jms.backend.BEXAResource.rollback(BEXAResource.java:1205) at weblogic.transaction.internal.ServerResourceInfo.rollback(ServerResourceInfo.java:1400) at weblogic.transaction.internal.ServerResourceInfo.rollback(ServerResourceInfo.java:664) at weblogic.transaction.internal.ServerSCInfo.startRollback(ServerSCInfo.java:365) at weblogic.transaction.internal.ServerTransactionImpl.localRollback(ServerTransactionImpl.java:1521) at weblogic.transaction.internal.ServerTransactionImpl.globalRollback(ServerTransactionImpl.java:2142) at weblogic.transaction.internal.TransactionImpl\$1.execute(TransactionImpl.java:1656) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:211) at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:198) weblogic.jms.common.JMSEException: Only one thread may use a JMS Session at a time. at weblogic.jms.frontend.FESession.rollbackAfterRecover(FESession.java:1773) at weblogic.jms.frontend.FESession.recover(FESession.java:1373) at weblogic.jms.frontend.FESession.invoke(FESession.java:2253) at weblogic.jms.dispatcher.Request.wrappedFiniteStateMachine(Request.java:609) at weblogic.jms.dispatcher.DispatcherImpl.dispatchSyncNoTran(DispatcherImpl.java:339) at weblogic.jms.client.JMSSession.recoverGuts(JMSSession.java:868) at weblogic.jms.client.JMSSession.rollback(JMSSession.java:632) at com.ncr.crm.framework.jms.DefaultJmsQueueReceiver.rollback(DefaultJmsQueueReceiver.java:184) at com.ncr.crm.framework.jms.DemoMessageConsumer.run(JmsMemoryLeakDemo.java:163) at java.lang.Thread.run(Thread.java:479) ----- Linked Exception ----- weblogic.jms.common.JMSEException: Only one thread may use a JMS Session at a time. at weblogic.jms.frontend.FESession.recover(FESession.java:1219) at weblogic.jms.frontend.FESession.invoke(FESession.java:2253) at weblogic.jms.dispatcher.Request.wrappedFiniteStateMachine(Request.java:609) at weblogic.jms.dispatcher.DispatcherImpl.dispatchSyncNoTran(DispatcherImpl.java:339) at weblogic.jms.client.JMSSession.recoverGuts(JMSSession.java:868) at weblogic.jms.client.JMSSession.rollback(JMSSession.java:632) at com.ncr.crm.framework.jms.DefaultJmsQueueReceiver.rollback(DefaultJmsQueueReceiver.java:184) at...</pre>	7.0 SP3 8.1 SP1
CR105403	<p>When trying to deploy an MDB on a stand-alone server, deployment fails if the license does not have the "Clustered JMS" component. License checking has been corrected.</p>	8.1 SP1

## JNDI

Change Request Number	Description	Release Fixed
CR091837	<p>Some customers had problems with the amount of time RMI stubs took to failover, if the destination server became unavailable. This amount of time varied with the operating system and the operating system version.</p> <p>This problem occurred because the socket creation from a client was taking too long to fail.</p> <p>This problem was resolved by providing the <code>-Dweblogic.client.socket.ConnectTimeout</code> flag, which sets the <code>Socket.connect()</code> method timeout in seconds.</p>	8.1 SP1
CR101433	<p>A deadlock occurred in the serialization of JNDI results.</p> <p>This problem occurred because the hashtable containing environment arguments (which is used to create the initial context) was being shared by different threads.</p> <p>This problem was resolved by using a workaround for a Sun bug, which involved cloning the hashtable containing environment arguments so that the hashtable is not shared by different threads.</p>	8.1 SP1
CR103711	<p>When one-phase deployment was used for an application, all <code>ejb-ref</code> lookups from within that application were resulting in a JNDI lookup. To improve performance, the <code>ejb-ref</code> values are now cached.</p>	8.1 SP1
CR105462	<p>When obtaining a new initial context via <code>WLInitialContextFactory</code> and then closing it repeatedly, a memory leak occurred in the JNDI client, and an Out of Memory exception was thrown.</p> <p>This problem occurred because the <code>TransactionHelper</code> class has a stack of <code>TransactionHelperImpl</code> objects. Objects were pushed into the stack when creating new contexts, but were not popped when the context was closed.</p> <p>This problem was resolved by modifying the code to pop the objects from the <code>TransactionHelper</code> when a context is closed.</p>	8.1 SP1

# JTA

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR097013	<p>A subcoordinator server was not recovering in-doubt transactions automatically after server down, requiring manual rollback or commitment by the DBA.</p> <p>Research revealed that the subcoordinator code was losing track of remote participants after server restart. A checkpoint was added to the code to properly track remote participants.</p>	7.0 SP3 8.1 SP1
CR099830	<p>JTA Migration sometimes failed with a <code>NullPointerException</code> or <code>ConnectException</code> after a transaction was propagated among two servers in cluster.</p> <pre>NullPointerException &lt;Feb 28, 2003 6:38:44 PM JST&gt; &lt;Warning&gt; &lt;JTA&gt; &lt;110213&gt; &lt;The activation of Transaction Recovery Service for server [serverB] fails. java.lang.NullPointerException at... ConnectException &lt;2003/02/27 13:11:20:JST&gt; &lt;Warning&gt; &lt;JTA&gt; &lt;110213&gt; &lt;The activation of Transaction Recovery Service for server [server1] fails.&gt; java.rmi.ConnectException: Destination unreachable; nested exception is: java.net.ConnectException: Connection refused: connect; No available router to destination at...</pre> <p>Research revealed that the code created a race condition between the notification of when the server was dead and when the information for that server was cleaned from the cache. If the notification that the server was dead came after the cache has been cleared, then the remaining Managed Server would get a <code>NullPointerException</code> when it tried to look for the location of the <code>tlog</code> of the dead Managed Server.</p> <p>A code fix resolved the problem.</p>	7.0 SP3 8.1 SP1
CR102358	<p>The following debug message was printed continuously every 5 seconds when JTARecovery debug was turned on:</p> <pre>&lt;Mar 29, 2003 1:09:15 PM EST&gt; &lt;Debug&gt; &lt;JTA&gt; &lt;BEA-110027&gt; &lt;ResourceDescriptor[JMSConnectionPool]: checkRecovery: &gt;</pre> <p>This message is no longer printed.</p>	8.1 SP1

Resolved Problems for Service Pack 1

Change Request Number	Description	Release Fixed
CR102400	<p>The transaction manager could generate a <code>NullPointerException</code> when trying to contact a remote server that is unavailable.</p> <p>This problem was resolved by a code change.</p>	8.1 SP1
CR102738	<p>An internal API, which was causing a <code>ServerMBean</code> lookup to fail, was changed. This change caused the Transaction recovery process to not recognize changes in a server's listen address.</p> <p>This problem was fixed. Additionally, code was added to ensure that asynchronous retries are attempted until new server information is obtained from the Administration Server. This ensures that the Transaction recovery process completes when the sub-coordinator server comes back up, without having to restart the coordinating server.</p>	7.0 SP3 8.1 SP1
CR103556	<p>When a <code>jndi.properties</code> file existed in the classpath that had a server URL pointing to a different server than the server being booted, a <code>javax.naming.NameAlreadyBoundException</code> was thrown when the transaction manager attempted to bind its <code>TransactionManager</code> object into the local JNDI tree. The exception was thrown because the bind operation was performed on the remote server's JNDI context in which a <code>TransactionManager</code> object already existed.</p> <p>This problem was resolved by a code change to ignore the <code>jndi.properties</code> file while binding objects into the local JNDI tree.</p>	8.1 SP1
CR103601	<p>If an object passed in the "equals" method of <code>weblogic.transaction.internal.XidImpl</code> was not an instance of <code>XidImpl</code> (for instance, it was of type <code>String</code>), then this exception is thrown:</p> <pre>java.lang.ClassCastException: java.lang.String at weblogic.transaction.internal.XidImpl.equals(XidImpl.java:114) at test.main(test.java:9)</pre> <p>The problem was solved with a code fix to check and report type mismatches.</p>	6.1 SP6 8.1 SP1

Change Request Number	Description	Release Fixed
CR106174	<p>If all but one participating resources in a transaction respond to prepare with the XA_RDONLY flag, the transaction manager should issue a one-phase commit to the pending participant. However, the TM was unnecessarily writing a commit record for this scenario.</p> <p>The code was changed so that, before writing a log record, it checks to see how many participants require 2nd phase; if one does, the write is skipped. This resolved the problem.</p>	7.0 SP3 8.1 SP1
CR106177	<p>While processing remote beforeCompletion callbacks, the TM resumes the transaction context prior to invoking the registered objects. However, statically registered resources are not enlisted when the transaction is resumed on the thread.</p> <p>The code was changed so that Enlistment is performed on such resources to allow beforeCompletion logic to perform additional updates as part of the transaction. This resolved the problem.</p>	7.0 SP3 8.1 SP1

## JVM

Change Request Number	Description	Release Fixed
CR101608	The JVM has a bug whereby it is possible for it to crash while reading an FVD-described class under IIOP. This problem was made worse by a bug in WebLogic Server, whereby it was possible to trivially provoke the problem. This CR refers to the WebLogic bug so that it is no longer trivially possible to cause this crash. However, the bug in the JVM still exists and is only fixed in J2SE 1.4.1_03 (and 1.3.1_09). BEA recommends upgrading to 1.4.1_03 when it becomes available.	8.1
CR105703	Starting in WebLogic Server 8.1, <code>JDriver</code> makes a new JDK 1.4 feature, <code>ByteBuffer</code> , to improve performance. <code>java.nio.charset.CharsetDecoder</code> is also used to decode the bytes in <code>ByteBuffer</code> that are returned from an Oracle server. Unfortunately, <code>CharsetDecoder</code> does not properly decode single byte data.  This problem was resolved by adding more international codesets to work around this JDK bug. The codesets include:  <code>Shift_JIS</code> , <code>SJIS</code> , <code>windows-31j</code> , <code>MS932</code> , <code>EUC_JP</code> , <code>EUC-JP</code> , <code>ISO-2022-JP</code> , <code>ISO2022JP</code> , <code>EUC_KR</code> , <code>EUC-KR</code> , <code>windows-949</code> , <code>MS949</code> , <code>EUC_CN</code> , <code>EUC-CN</code> , <code>windows-936</code> <code>MS936</code> , <code>GBK</code> , <code>Big5</code> , <code>windows-950</code> , <code>MS950</code> , <code>Big5_HKSCS</code> , <code>Big5-HKSCS</code>  These names are based on <a href="#">JDK 1.4 internationalization</a> .	8.1 SP1

## Localization

Change Request Number	Description	Release Fixed
CR108583	The <code>messageid="DefaultMigratableSuffix"</code> of <code>ja/JP/ManagementText.xml</code> was translated to a Japanese String, resulting in an inability for customers to delete a server that was created with the Configuration Wizard.  This problem occurred because this String, which is used in a String comparison, should not have been translated in the catalog.  This problem was resolved by removing the Japanese translation of the String.	8.1 SP1

## Node Manager

Change Request Number	Description	Release Fixed
CR093664	The comment at the beginning of <code>nodemanager.properties</code> that documents the syntax of this file only included SSL configuration information. It now includes descriptions for other commonly used properties.	8.1 SP1

## Plug-Ins

Change Request Number	Description	Release Fixed
CR096625	<p>The Apache plug-in sent a <code>X_WEBLOGIC_FORCE_JVMID</code> header to a WebLogic Server instance in a cluster. However, <code>X_WEBLOGIC_FORCE_JVMID</code> should only be sent when the server list is non-clustered and the current server does not yet have a JVM ID.</p> <p>This problem was resolved by modifying the code so that <code>X_WEBLOGIC_FORCE_JVMID</code> is only sent when the server list is non-clustered and the current server does not yet have a JVM ID.</p>	7.0 SP3 8.1 SP1
CR097202 CR103161	<p>A customer needed to determine a browser's cipher strength on the WebLogic side, in a configuration that included a plug-in with 128-bit step up certs between WebLogic Server and the browser. The browser had 40-bit or 128-bit strength. The customer used the following code to obtain the browser's cipher strength:</p> <pre>(Integer) httpRequest.getAttribute("javax.servlet. request.key-size")</pre> <p>However, the value returned was always 128 (although the browser was set to use a cipher size less than 128-bit).</p> <p>This problem occurred because <code>HTTPS_KEYSIZE</code> determines the strength of the connection that is made. Because 128-bit certs were used on the plug-in, 128 was returned, regardless of the strength used by the browser. When the request was directed to WebLogic Server, 40 or 128 was returned, depending on the browser strength. The plug-in does not modify the <code>HTTPS_KEYSIZE</code> or <code>HTTPS_SECRETKEYSIZE</code> headers.</p> <p>To meet this need, two new headers were implemented: <code>WL-Proxy-Client-Keysize</code> and <code>WL-Proxy-Client-Secretkeysize</code>. Both headers values can be obtained using <code>request.getHeader()</code>.</p>	6.1 SP5 8.1 SP1

Resolved Problems for Service Pack 1

Change Request Number	Description	Release Fixed
CR100361	<p>The <code>READ_ERROR</code> exception did not differentiate between reading from a client and reading from a back-end server. (In contrast, there were two different <code>WRITE_ERROR</code> exceptions: <code>WRITE_ERROR_TO_CLIENT</code> and <code>WRITE_ERROR_TO_SERVER</code>.)</p> <p>This problem was resolved by introducing the following new exception types:</p> <ul style="list-style-type: none"> <li>• <code>READ_ERROR_FROM_CLIENT</code>—error when reading data from client</li> <li>• <code>READ_ERROR_FROM_SERVER</code>—error when reading data from backend server</li> <li>• <code>READ_ERROR_FROM_FILE</code>—error when reading from temporary file which stores post data</li> <li>• <code>WRITE_ERROR_TO_FILE</code>—error when writing post data to the temporary file</li> </ul>	6.1 SP5 8.1 SP1
CR101222	<p>After fail-over from a primary server to a secondary server for NSAPI and Apache did not recover the failed primary server after the time specified in <code>MaxSkipTime</code> had elapsed.</p> <p>This problem occurred because the bad primary server had not been reset.</p> <p>This problem was resolved by resetting the primary server after the <code>MaxSkipTime</code> had elapsed.</p>	8.1 SP1
CR101596	<p>In the <code>ssl_certchain_verify_callback</code> method, <code>SSLNoErr</code> was returned as the error condition for some failure cases of certification validation. This included cases where the CA certificate had basic constraints that were not marked as critical and the strict setting was enabled (instead of the default "strong" setting), and if the certificate was not marked as being a CA.</p> <p>This problem was resolved by returning <code>X509CertChainInvalidErr</code> for these failure cases (instead of <code>SSLNoErr</code>).</p>	6.1 SP5 8.1 SP1
CR102616	<p>The NSAPI plug-in now supports dynamic DNS lookups on a DNS name when the name returns a list of IP addresses from the DNS server and the plug-in is configured with <code>WebLogicHost = 'DNS name'</code>.</p>	7.0 SP3 8.1 SP1

## RMI/RMI-IIOP

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR095810	<p>You could not use callbacks through a firewall when using the thin client <code>.jar</code> files (<code>wlclient.jar</code> and <code>wljmsclient.jar</code>) and IIOP or IIOPS. Callbacks did work when using tunneling.</p> <p>Since Bi-Directional IIOP was already supported in the thin-client when tunneling was used, this problem was resolved by generalizing the code so that BiDir could be used with standard IIOP as well as over HTTP.</p>	8.1 SP1
CR097787	<p>Although thin-client IIOP implemented an asynchronous "PeerGone" mechanism, this mechanism did not wakeup pending requests with an exception when a network partition occurred.</p> <p>This problem was resolved by modifying the code so that when BiDir is used, the "PeerGone" mechanism signals an exception on the socket.</p>	8.1 SP1
CR101776	<p>There were no dynamic debug flags for thin clients.</p> <p>This problem was resolved by adding the following debug flags for thin clients:</p> <ul style="list-style-type: none"><li>• <code>-Dweblogic.debug.client.iiop=true</code></li><li>• <code>-Dweblogic.debug.client.startup=true</code></li><li>• <code>-Dweblogic.debug.client.http=true</code></li><li>• <code>-Dweblogic.debug.client.security=true</code></li><li>• <code>-Dweblogic.debug.client.ots=true</code></li><li>• <code>-Dweblogic.debug.client.cluster=true</code></li><li>• <code>-Dweblogic.debug.client.ssl=true</code></li><li>• <code>-Dweblogic.debug.client.dgc=true</code></li></ul>	8.1 SP1
CR102875	<p>When attempting to deploy from ant (<code>wldeploy</code>) on a WebLogic Server instance using the IIOP protocol, the server produced a <code>NullPointerException</code> in <code>weblogic.corba.utils.MarshaledString</code>.</p> <p>This problem occurred because WebLogic was erroneously returning null when retrieving a repository ID from a stub that had been cached by a previous lookup.</p> <p>This problem was resolved by initializing the type ID to the result of the <code>info.getRepositoryId</code> method in cases where the type ID is null.</p>	8.1 SP1

## Resolved Problems for Service Pack 1

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR103388	WebLogic RMI now supports arrays of primitives.	7.0 SP3 8.1 SP1
CR105892	When using JMS with the IIOP protocol, abnormal disconnection by a client was not being properly detected in the server. This causes problems with stateful information associated with the client, like the JMS client identifier.  Abnormal disconnection is now detected and handled appropriately.	8.1 SP1
CR106521	Customer could not pass Any's containing enums to or from CORBA servers.  This problem was resolved by fixing the marshaling of enums inside Any's.	8.1 SP1
CR106750	Client-server applications with clients connecting to WebLogic Server over IIOP and invoking server-side logic through EJBs may have experienced intermittent IIOP errors.  Specifically, these were CORBA/IIOP MARSHAL and indirection errors in the <code>read_value()</code> method during thin client access.  This problem was resolved with a code fix.	8.1 SP1
CR108332	When using a thin client ( <code>wlclient.jar</code> ), customers may have encountered a marshalling exception that indicated there had been an IIOP stream corruption. This problem occurred when IIOP or T3 was specified as the protocol in multiple minor versions of J2SE 1.4, but not when <code>weblogic.jar</code> was in the client classpath.  This problem occurred because of issues with the encoding and decoding of custom marshaled valuetypes and in the way WebLogic Server kept track of indirections when marshaling valuetypes.  These issues have been resolved.	8.1 SP1
CR109032	Accessors with checked exceptions were not mapped properly in the IDL, resulting in the wrong IDL being generated.  This problem was resolved by modifying the IIOP name mangling code so that it now checks whether accessors ( <code>getFoo()</code> ) and mutators ( <code>setFoo()</code> ) throw checked exceptions. If they do, then they are not mangled to IDL attributes (that is, just treated as standard functions).	8.1 SP1

# Security

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR097038	When using the Messaging Bridge, non-trusted domains only worked when using synchronous mode communications.	7.0 SP3
CR102656	This problem has been resolved so that non-trusted domains also work when using asynchronous mode communications.	8.1 SP1
CR098843	The <code>weblogic.management.runtime.ExecuteThread.getUser</code> method always returned <code>null</code> .  This problem occurred because there was no way to obtain the current user for a given thread, and was resolved by adding code that does this.	8.1 SP1
CR100783	The <code>ImportPrivateKey</code> utility produced the following message when the wrong private key password was entered: <code>ASN.1: Lengths longer than 32 bits are not supported.</code> That is, rather than indicating a problem with the password, the error message described a problem with the key length.  This problem was resolved by modifying the error message to be more appropriate.	8.1 SP1
CR101531	The memory usage of the Embedded LDAP server was reduced by decreasing the size of a buffer used for logging. This reduced the total memory usage by 1.8 million bytes.	8.1 SP1
CR101770	The <code>SSL Login Timeout</code> attribute was unintentionally dropped from the WebLogic Server Administration Console. Support for the attribute was added back into the Administration Console.	8.1 SP1
CR101785	A <code>UsernamePasswordLoginModule LoginModule</code> was added for use by the IIOP thin-client. This class has the same API as that of the <code>weblogic.security.auth.login.UsernamePasswordLoginModule</code> class, but in this implementation, there is a new property called <code>authOnLogin</code> . When true, this property forces the <code>login</code> method to perform authentication (rather than on the first invocation). The default for this new property is false.	8.1 SP1

Resolved Problems for Service Pack 1

Change Request Number	Description	Release Fixed
CR102164	<p>The MBean implementations for the WebLogic Authentication provider and LDAP Authentication providers threw a <code>weblogic.management.utils.InvalidPasswordException</code> if the password was not valid. This behavior was inconsistent with the <code>weblogic.management.security.authentication.UserPasswordEditor</code>'s <code>changeUserPassword</code> and <code>resetUserPassword</code> methods, which do not declare a <code>InvalidPasswordException</code> exception.</p> <p>This problem was resolved by removing the <code>weblogic.management.utils.InvalidPasswordException</code> and changing the Authentication providers' MBean implementations to throw a <code>weblogic.management.InvalidParameterException</code> when a password is not valid.</p>	8.1 SP1
CR102221	<p>The SSL <code>closeWriteHandler</code> method produced an <code>IOException</code> when the output stream had already been closed by an application.</p> <p>This problem occurred because the <code>closeWriteHandler</code> method did not check whether the <code>WriteHandler</code> was already closed, and attempted to write an alert to the output stream.</p> <p>This problem was resolved by adding this check.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR102251	<p>In 6.x, the t3 protocol sends authenticated user object between clients and servers. In 7.x and higher, the t3 protocol sends authenticated subject object (which extends the authenticated user object) between clients and servers. When a 7.x server communicates with a 6.x client, it converts any authenticated subjects back to authenticated users before sending them to a client.</p> <p>If the client was communicating with both 8.1 servers and 6.1 servers, it received an authenticated user when it created a context to a 6.x server. When it created a context to a 8.x server, it received an authenticated subject. Depending upon which context was created last, an authenticated subject or authenticated user would be sent to the client.</p> <p>If the context to the 8.1 server was created last, then the client identity was an authenticated subject, which would be passed to both the 6.1 and 8.1 servers. When the authenticated subject was passed to the 6.x server, the 8.1 client attempted to convert it into an authenticated user. The code to do this was only intended to be used on the server-side, so it was failing with an exception that eventually caused an assertion failure. When run with a 6.1 client, things worked correctly, as the client always received an authenticated user.</p> <p>This problem was resolved by modifying the code so that conversions from an authenticated subject to an authenticated user can be called on the client side.</p>	8.1 SP1
CR102443	<p>Attempts to use one WebLogic Server instance as a LDAP server to another WebLogic Server instance that had a LDAP Authentication provider configured failed. The server acting as the LDAP server threw a <code>NullPointerException</code>, while the server configured to use external LDAP hung.</p> <p>This problem was occurred because user and group DN's did not match the hierarchy in the embedded LDAP server for the other domain.</p> <p>This problem was resolved by modifying the code so that an informative error is returned when the credentials supplied are invalid.</p>	8.1 SP1
CR104191	<p><code>URLResource</code> converted the context path to lowercase on Windows systems, causing a case mismatch between the context path used by Web applications and the WebLogic Security Framework when re-deploying a Web application with security constraints. As a result, security constraints that were removed from the Administration Server were not be propagated to the Managed Servers in the domain.</p> <p>This problem was resolved by adding a verification to ensure that the case of both context paths match.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR104502	<p><code>utils.der2pem</code> wrote out an extra new line if the number of characters was an exact multiple of the line width, resulting in an invalid <code>.pem</code> file.</p> <p>This problem has been resolved.</p>	8.1 SP1
CR104713	<p>When a WebLogic Server instance was configured for two-way SSL and a WebLogic Identity Assertion provider was configured with the <code>AuthenticatedUserX.509</code> token type, a <code>ClassCastException</code> was thrown while the server was authenticating the client connecting over SSL. This exception indicated that there was a problem with <code>DefaultIdentityAsserterProviderImpl</code> (the Weblogic Identity Assertion provider).</p> <p>This problem occurred because a token of the wrong type was being passed to the WebLogic Identity Assertion provider.</p> <p>This problem was resolved by ensuring that an array of certificates is passed to the WebLogic Identity Assertion provider.</p>	8.1 SP1
CR105809	<p>Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisorienotifications/BEA03-33.jsp">http://dev2dev.bea.com/resourcelibrary/advisorienotifications/BEA03-33.jsp</a>.</p>	7.0 SP3 8.1 SP1
CR106027	<p>When logging into the Administration Console as a <code>Monitor</code>, a user could still have access to the 'Configure new Application' screen. This constituted a security hole, because it allowed a monitor to upload a file to any place the WebLogic server can see. The problem was resolved with a code fix.</p>	7.0 SP3 8.1 SP1
CR108958	<p>Currently WebLogic Server allows a JCE provider to be configured and used for SSL unless it is determined that the JCE provider is not working with Certicom. If the latter is the case, WebLogic Server adds it to the list of JCE providers known to cause problems. When Certicom discovers that the installed JCE provider is in this list, it ignores it and uses the <code>Jsafe</code> implementation instead.</p> <p>The IBM JCE provider has been added to this list, so it is now ignored by the Certicom SSL implementation, but is still available for other applications.</p>	8.1 SP1

## Servlets & JSPs

Change Request Number	Description	Release Fixed
CR088670	<p>When returning a license error, WebLogic Server produces a 403 page that explains the problem. However, this message has a content-type of <code>x-weblogic/internal</code>, when it should have a content-type of <code>text/html</code>. This caused an "open with application or download file" prompt on Mozilla.</p> <p>This problem was resolved by modifying the content-type to be <code>text/html</code>.</p>	8.1 SP1
CR092778	<p>When using JISAutoDetect encoding (which is used to accept Shift-JIS, EUC-JP and ISO-2022-JP for input streams without code/configuration modification) with an HTTP request, an <code>UnsupportedEncodingException</code> was thrown, starting with the following lines:</p> <pre>java.io.UnsupportedEncodingException: JISAutoDetect at sun.io.Converters.getConverterClass(Converters.java:102) ) at sun.io.Converters.newConverter(Converters.java:133) at sun.io.CharToByteConverter.getConverter(CharToByteConverter.java:62) at weblogic.servlet.internal.ServletRequestImpl.setCharacterEncoding(ServletRequestImpl.java:344) ... </pre> <p>This problem occurred when a JSP/servlet called <code>request.setCharacterEncoding("JISAutoDetect")</code>.</p> <p>This problem was caused because <code>ServletRequestImpl.java</code> used <code>sun.io.CharToByteConverter</code>, instead of <code>sun.io.ByteToCharConverter</code>. <code>CharToByteConverter</code> is for an output converter, while <code>JISAutoDetect</code> is only available for an input stream.</p> <p>This problem was resolved by changing to <code>ByteToCharConverter</code>, which accepts <code>JISAutoDetect</code>.</p>	6.1 SP5 8.1 SP1

Change Request Number	Description	Release Fixed
CR093209	<p>A new installation of WebLogic Server threw a <code>NullPointerException</code> when a successfully deployed Web application with a property set to <code>java</code> protocol was accessed. The browser returned Error 500--Internal Server Error. The problem did not occur with an upgrade installation.</p> <pre data-bbox="266 552 1026 1112">&lt;Dec 16, 2002 11:59:05 AM PST&gt; &lt;Error&gt; &lt;HTTP&gt; &lt;[WebAppServletContext(2092664,sampleApp,/sampleApp)] Servlet failed with Exception java.lang.NullPointerException at weblogic.servlet.JSPServlet.service(JSPServlet.java:132) ) at javax.servlet.http.HttpServlet.service(HttpServlet.java:853) at weblogic.servlet.internal.ServletStubImpl.invokeServlet (ServletStubImpl.java:262) at weblogic.servlet.internal.ServletStubImpl.invokeServlet (ServletStubImpl.java:198) at weblogic.servlet.internal.WebAppServletContext.invokeSe rvlet(WebAppServletCon ext.java:2637) at weblogic.servlet.internal.ServletRequestImpl.execute(Se rvletRequestImpl.java:235) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.jav a:139) at weblogic.kernel.ExecuteThread.run(ExecuteThread.java:12 0)&gt;</pre> <p>Analysis revealed an error in <code>ZipSource.getURL()</code>. The problem was resolved by adding logic to check for a null value of <code>source.getURL()</code>.</p>	6.1 SP5 8.1 SP1
CR093625	<p>In WebLogic Server 6.1 SP3 and SP4, servers misdirected output for JSP includes in JSP pages. The problem was resolved with changes to the servlet engine to unify handling of includes, forwards, wrapped-responses, and JSP BodyTags.</p>	6.1 SP5 8.1 SP1
CR094190	<p>According to the JVM specification, the size limit for a method is 64K. Some WebLogic Server customers have JSPs with many body tags, for which the generated Java code exceeded the 64K limit for the <code>__jspService</code> method.</p> <p>The problem was solved by a code change. Now, when an empty <code>BodyTag</code> of the form <code>&lt;my:tag ... /&gt;</code> is encountered, the generated code that would normally set up scope for the body is replaced by a single call to a static method in <code>StandardTagLib.java</code>. This tricks the <code>BodyTag</code> into thinking it is being invoked as usual from the JSP source.</p>	6.1 SP5 8.1 SP1

Change Request Number	Description	Release Fixed
CR094488	<p>A new capability was added to allow a user to securely access HTTPS resources in a session that was initiated using HTTP, without loss of session data. To enable this new feature, add <code>AuthCookieEnabled="true"</code> to the <code>WebServer</code> element in <code>config.xml</code>:</p> <pre data-bbox="327 548 1063 569">&lt;WebServer Name="myserver" AuthCookieEnabled="true" /&gt;</pre> <p>This will cause a new secure cookie to be sent to the browser when authenticating via an HTTPS connection. Once set, the session can access other security-constrained HTTPS resources only if the cookie is sent from the browser.</p> <p><b>Note:</b> If authenticating via plain HTTP, the secure cookie will not be set or required for any HTTPS resources. When accessing a non-protected HTTPS resource, the cookie will not be verified (since it will not have been sent from the browser). This allows the browser to access non-protected HTTPS resources without the user logging in.</p>	6.1 SP5 8.1 SP1
CR095981	<p><code>&lt;charset-mapping&gt;</code> in <code>weblogic.xml</code> was ignored when compiling JSP files with <code>precompile=true</code>.</p> <p>As a result, some of double byte characters were garbled because of the mismatch between the character's encoding specified in <code>&lt;charset-mapping&gt;</code> and in the compiled classes.</p> <p>Analysis revealed an error in the precompiler. The problem was solved with a code fix.</p>	6.1 SP6 8.1 SP1

Resolved Problems for Service Pack 1

Change Request Number	Description	Release Fixed
CR097719	<p>In WebLogic Server SP02, SP03, and SP04, when getting certain POST requests from WAP devices, a Web application encountered a <code>java.util.ConcurrentModificationException</code>:</p> <pre data-bbox="266 522 1032 770">&lt;29.jan.03 13:03:05 WET&gt; &lt;Error&gt; &lt;HTTP&gt; &lt;[WebAppServletContext(2810713,TnMFF,/TnMFF)] Servlet failed with Exception java.util.ConcurrentModificationException at java.util.HashMap\$HashIterator.next(HashMap.java:736) at weblogic.utils.enumerations.IteratorEnumerator.nextElement(IteratorEnumerator.java:25) at com.colibria.core.xmlswitch.XMLSwitch.getQueryStringXML(XMLSwitch.java:347) at ...</pre> <p>Analysis revealed that when a charset is associated with the request's <code>content-type</code>, the code read the data again using the encoding, and reset the query parameters, in the application. The application already has the enumeration object and iterates it (<code>request.getParameterNames</code>), and then tries to get the value of the parameter (<code>request.getParameterValues(k)</code>). At that point, the query parameters are wiped out and <code>getParameterValues</code> method tries to set it, resulting in the exception.</p> <p>The problem was resolved by a code change to set query parameters after they are wiped, so that the data can be read again when a charset is associated with the request's <code>content-type</code>.</p>	8.1 SP1 6.1 SP5
CR097759	<p>There was no timestamp associated with VM thread dumps.</p> <p>This problem was resolved by modifying the code so that a message of the following format is printed before a VM thread dump (applies only to thread dumps taken with the <code>weblogic.Admin</code> thread dump command):</p> <pre data-bbox="266 1255 1032 1277">THREAD DUMP from JVM taken at 'Mon Mar 24 16:26:13 2003'</pre>	8.1 SP1
CR098955	<p>If <code>logRotationBeginTime</code> was a past date and the interval between <code>logRotationBeginTime</code> and today's date was large, calculation of the rotation time was flawed, due to improper casts of long to int. This has now been resolved.</p>	5.1 SP14 8.1 SP1

Change Request Number	Description	Release Fixed
CR100068 CR110324	<p>You could not use JSTL tags in JSPs that included Japanese characters. When the JSP was executed, an error starting with the following lines would occur:</p> <pre>java.io.IOException: javax.servlet.jsp.JspException: The taglib validator rejected the page: "org.xml.sax.SAXParseException: An invalid XML character (Unicode: 0x82) was found in the CDATA section." ... </pre> <p>This problem occurred because of the following, conflicting conditions:</p> <ol style="list-style-type: none"> <li>1. The page encoding of JSPs is defined by <code>Shift_JIS</code>, as in:  <pre>&lt;%@ page pageEncoding="Shift_JIS" %&gt;</pre></li> <li>2. Using multibyte characters (Japanese) in JSPs.</li> <li>3. Using JSTL tags.</li> </ol> <p>This problem was resolved by modifying the encoding of the byte stream returned from the <code>weblogic.servlet.jsp.StandardTagLib.makeXMLStream</code> method to UTF-8.</p>	7.0 SP3 8.1 SP1
CR100172	<p>When multiple chunked requests were posted to a servlet, a number of the requests failed with a <code>NumberFormatException</code>, starting with the following lines:</p> <pre>&lt;ExecuteThread: '11' for queue: 'default'&gt; &lt;kernel identity&gt; &lt;&gt; &lt;101017&gt; &lt;[ServletContext(id=4864139,name=387551,context-path=)] Root cause of ServletException&gt; java.lang.NumberFormatException: ... </pre> <p>This problem occurred because the code incorrectly attempted to detect the end of the chunk at the wrong place.</p> <p>This problem was resolved by modifying the code so that it reads until the end of the stream, and detects the end of the chunk correctly.</p>	7.0 SP3 8.1 SP1
CR100188	<p>The <code>AuthCookie</code> feature did not work for certificate authentication. (<code>AuthCookie</code> is a secure cookie used to make HTTP to HTTPS transitions secure.) This capability was added in WebLogic Server 6.1 SP5 to allow a user to securely access HTTPS resources in a session that was initiated using HTTP, without loss of session data.</p> <p>This problem was resolved with a code fix.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR100260	<p>When an HTTP request was proxied by a WebLogic proxy to a cluster using the non-default network channel, the cluster list returned was with respect to the default channel. Thereafter, the plug-in started proxying to the default channel, instead of the original network channel, which may not have been reachable.</p> <p>This problem was resolved by modifying the code so that the dynamic server list is aware of non-default network channels.</p>	8.1 SP1
CR100570	<p>While saving changes for <code>weblogic.xml</code> from WebLogic Builder, it created extraneous entries in the <code>weblogic.xml</code> file for the deployment descriptor MBeans. The extra entries were defaults and did not need to be added to the <code>weblogic.xml</code> file.</p> <p>This problem was resolved by modifying the code so that the entries are only written to the <code>weblogic.xml</code> file if the default values have been changed.</p>	8.1 SP1
CR101996	<p>Use of <code>xmlx</code> tags with request and response wrappers resulted in <code>ClassCastException</code>s since the tags internally cast to the internal request and response implementations. This problem was resolved by unwrapping the request and response instances to the original internal representation.</p> <p>Use of the <code>xmlx</code> tag may have also produced an <code>IllegalStateException</code> indicating that an attempt was made to use <code>getWriter</code> after <code>getOutputStream</code>. This problem was resolved by modifying the code to appropriately reset the output state prior to calling <code>getOutputStream</code>.</p>	8.1 SP1
CR102057	<p>Including a static resource in a request with no buffer size (for example, <code>response.setBufferSize(0)</code>) could lead to an infinite loop. This problem was resolved by guarding against this case.</p>	8.1 SP1
CR102062	<p>The <code>ServletContext.getResourcePaths</code> method returned incorrect results when called in a WAR file, and in the case of split-directory deployments, the <code>Set</code> returned by this method only contained the results from the <code>srcdir</code> (it did not contain the results from the <code>outputdir</code>).</p> <p>This problem was resolved by a code fix.</p>	8.1 SP1
CR102077	<p><code>weblogic.servlets.internal.InternalRequestDispatcher</code> throws a <code>StringIndexOutOfBoundsException</code> if given a URI whose length is 0 (for example, an empty HTML file).</p> <p>This problem was resolved by guarding against this condition.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR102499	<p>If an attribute was added to the request using the Web application classloader and if the request gets delegated to a child classloader of the Web application's classloader, upon <code>request.getAttribute()</code>, the Web application container was trying to deserialize the attribute in order to avoid a <code>ClassCastException</code>. This was resulting in Serialization errors.</p> <p>The Web application container code will now be able to detect that the class was loaded by a parent classloader. If true, it will not deserialize.</p> <p>However, if the child places attributes into the request using <code>setAttribute()</code>, this will not work if</p> <ul style="list-style-type: none"> <li>• the class or references within are child scoped only, or</li> <li>• the class or references within are not serializable.</li> </ul> <p>You can workaround the second problem by pushing and popping the Web application classloader temporarily for <code>setAttributes()</code> in the child classloader.</p>	8.1 SP1
CR102667	<p>A <code>NullPointerException</code> was thrown by a Web application module if the context with the same name was previously deployed using the old deployment model (6.x deployment model).</p> <p>This problem was resolved by ensuring that if present, the old deployment model is handled correctly.</p>	8.1 SP1
CR102763	<p><code>ExternalDNSName</code> is deprecated for Network Channels (not servers). Customers should make use of the new Public Address for channels, when a firewall is between a proxy Web server and a cluster. Additionally, customers should use paired channels for HTTP and HTTPS. More information is available in the "<a href="#">Firewall Between Proxy Layer and Cluster</a>" section of <i>Using WebLogic Server Clusters</i>.</p>	8.1 SP1
CR103059	<p>A <code>&lt;url-pattern&gt;</code> value with a single character in a <code>&lt;servlet-mapping&gt;</code> element was not honored in a <code>web.xml</code> deployment descriptor. For example, <code>&lt;url-pattern&gt;*.f&lt;/url-pattern&gt;</code> produced a 404 File Not Found error when attempting to access <code>welcome.f</code>, but <code>&lt;url-pattern&gt;*.oop&lt;/url-pattern&gt;</code> worked properly when attempting to access <code>welcome.oop</code>.</p> <p>This problem has been resolved so that single character <code>&lt;url-patterns&gt;</code> now work as expected.</p>	8.1 SP1

Resolved Problems for Service Pack 1

Change Request Number	Description	Release Fixed
CR103095	<p>When the authentication method associated with a Web application module was invalid and the Web application module was being deployed, no error message was produced, and the authentication method was defaulted to BASIC authentication without notification.</p> <p>This problem was resolved by modifying the code so that a Web application module with an invalid authentication method will not be deployed, and a message indicating the problem will be displayed in the server log file, as well the Administration Console or stdout (depending on how the Web application module is deployed).</p>	8.1 SP1
CR103192	<p>If clients did not have session cookies enabled, users were not able to log in to secured Web sites over HTTPS. It was possible to workaround this problem by manually disabling the <code>AuthCookie</code>, which up until WebLogic Server 8.1, had been disabled by default.</p> <p>This problem was resolved by automatically disabling the <code>AuthCookie</code> check when the Web application has session cookies disabled.</p>	8.1 SP1
CR103247	<p>WebLogic Server could not locate implicit TLDs when using the split directory feature.</p> <p>This problem occurred because the <code>resourceFinders</code> were being updated after the implicit TLDs were loaded. (The updating of <code>resourceFinders</code> is responsible for adding the <code>sredir</code> and the destination (<code>outputdir</code>) directories.)</p> <p>This problem was resolved by updating the <code>resourceFinders</code> before the extraction process.</p>	8.1 SP1
CR103925	<p>The <code>setAttribute</code> method only checked for <code>hashCode</code> equality. It now also checks the result of the <code>equals</code> method for the old and new objects.</p>	8.1 SP1
CR104410	<p>When <code>PersistentStoreType</code> was set to "replicated," customers who had successfully deployed an application received an "Internal Server error" message when attempting to access the application from a browser. In addition, the following exception was logged on the server:</p> <pre data-bbox="266 1364 1022 1442">weblogic.management.NoAccessRuntimeException: Access not allowed for subject: principals=[], on ResourceType: ServletSessionRuntime Action: register, Target: null</pre> <p>This problem was resolved by modifying the code so that a session with session monitoring enabled could be successfully created.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR104469	The <code>fastfile.dll</code> file, which is required for native IO for the <code>FileServlet</code> on Windows, was not working with the JRockit JVM. This problem has been resolved.	8.1 SP1
CR105171	'load-on-startup' JSP files were being compiled by <code>weblogic.jspc</code> , even when the <code>&lt;jspServlet&gt;</code> element in <code>weblogic.xml</code> indicated that <code>weblogic.servlet.WlwJSPServlet</code> should be used.  This problem was resolved so that the specified compiler will be used.	8.1 SP1
CR106172	The deployment of a Web application with the <code>precompile</code> option set to <code>true</code> in <code>weblogic.xml</code> failed, with a JSLT tag validator error of <code>org.xml.sax.SAXParseException</code> .  This problem occurred because when <code>precompile</code> was set to <code>true</code> , the JSP file was not being passed to the method that performs validation.  This problem was resolved by modifying the code to properly pass the JSP file to the method that performs validation.	8.1 SP1

## System Administration

Change Request Number	Description	Release Fixed
CR080353	<p>Converting the <code>weblogic.properties</code> file using the domain convertor does not produce any entry for <code>weblogic.password.guest=&lt;guestpassword&gt;</code>. Although the conversion is successful, an error similar to the following may be encountered:</p> <pre>java.lang.SecurityException: Authentication for user guest denied in realm weblogic</pre> <p>This problem was resolved by modifying the code so that a warning indicating how to obtain the old behavior is displayed:</p> <pre>&lt;Apr 12, 2003 10:29:30 AM EDT&gt; &lt;Warning&gt; &lt;Management&gt; &lt;BEA-141186&gt; &lt;User Guest has been disabled by default. If you would like to enable Guest, you should do it manually.&gt;</pre>	8.1 SP1
CR098920	<p>Improvements were made to message catalog Detail/Cause/Action information, based upon BEA Support's review of existing System Administration catalog messages.</p>	8.1 SP1
CR099404	<p>When using the Security Service Provider Interfaces (SSPIs) to create MBeans to manage your custom security providers, if you attempted to use the same <code>ObjectName</code> for more than one MBean, WebLogic Server wrapped the underlying JMX exception, <code>InstanceAlreadyExistsException</code>, inside a generic <code>MBeanException</code>. In SP1, the true exception is thrown.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR100331	<p>When starting up a server in an empty directory using the following command, the warning message transposed the Domain Name and Server Name:</p> <pre data-bbox="327 517 1089 765"> .  /bea/wls810/curr2weeks/weblogic/server/bin/setWLSEnv.sh  /bea/wls810/curr2weeks/jdk141_02/bin/java  -Dweblogic.management.GenerateDefaultConfig=true  -Dweblogic.ListenAddress=172.18.136.38  -Dweblogic.ListenPort=7101  -Dweblogic.Domain=rmiioopStressCluster  -Dweblogic.Name=adminServer  -Dweblogic.management.password=weblogic  -Dweblogic.management.username=weblogic weblogic.Server </pre> <p>The incorrect warning message was similar to:</p> <pre data-bbox="327 822 1089 899"> &lt;Mar 6, 2003 9:17:24 AM EST&gt; &lt;Warning&gt; &lt;Management&gt; &lt;BEA-140012&gt; &lt;Creating default domain <b>adminServer</b> with default server <b>rmiioopStressCluster</b>.&gt; </pre> <p>This problem was resolved by correcting the warning to:</p> <pre data-bbox="327 956 1089 1034"> &lt;Mar 6, 2003 9:17:24 AM EST&gt; &lt;Warning&gt; &lt;Management&gt; &lt;BEA-140012&gt; &lt;Creating default domain <b>rmiioopStressCluster</b> with default server <b>adminServer</b>.&gt; </pre>	8.1 SP1
CR101002	<p>Security roles for an EAR that were only defined in application.xml (that is, not also defined in weblogic-application.xml) were not made available to the WebLogic Security Service, and therefore were not visible in the Administration Console.</p> <p>This problem was resolved with a code fix, and these resources now appear in the Administration Console as expected.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR101144	<p>When attempting to log into UDDI after a server reboot, the following exception was thrown:</p> <pre data-bbox="266 487 1029 739">&lt;Mar 25, 2003 11:23:05 AM CST&gt; &lt;Error&gt; &lt;HTTP&gt; &lt;BEA-101017&gt; &lt;[ServletContext(id=1 4133705,name=uddi,context-path=/uddi)] Root cause of ServletException. weblogic.management.NoAccessRuntimeException: Access not allowed for subject: principals=[], on ResourceType: EmbeddedLDAPConfig Action: read, Target: Credential at weblogic.management.internal.SecurityHelper\$IsAccessAll owedPrivilegeA ction.wlsRun(SecurityHelper.java:492)</pre> <p>The exact steps to reproduce this problem are as follows:</p> <ol data-bbox="266 791 969 965" style="list-style-type: none"> <li>1. In an empty directory, create and start a WebLogic Server instance.</li> <li>2. Load <code>http://localhost:7001/uddiexplorer/index.jsp</code>.</li> <li>3. Navigate to Search Private Registry and click SEARCH.</li> <li>4. Stop the server, then restart it.</li> <li>5. Repeat steps 2 and 3 above, and the error will occur.</li> </ol> <p>This problem is due to a change to ensure that access to protected attributes is not allowed unless users are granted the Admin security role.</p> <p>This problem was resolved by modifying the UDDI code to run as an identity that has administrative privileges. This way, it can obtain access to the embedded LDAP password attribute it requires.</p>	8.1 SP1
CR101246	<p>For any action that requires reading an encrypted password field (such as attempting to deploy a JDBC connection pool), users of the Administration Console who are granted the <code>Deployer</code> security role received an exception similar to the following:</p> <pre data-bbox="266 1260 1029 1338">Access not allowed for subject: principals=[DeployerUser, Deployers], on ResourceType: JDBCConnectionPool Action: read, Target: Password</pre> <p>This problem was resolved by modifying the code so that users granted the <code>Deployer</code> security role have read access on all the <code>DeploymentMBeans</code>. This means that users granted the <code>Deployer</code> security role will also have read access to encrypted attributes on <code>DeploymentMBeans</code>. The password attribute on the <code>JDBCConnectionPoolMBean</code> was also affected by this change.</p>	8.1 SP1

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR102149	<p>A WebLogic Server instance gave an <code>OutOfMemoryError</code> when receiving 21,000 persisted messages synchronously in a transacted session on Solaris.</p> <p>The garbage collector was slow at reclaiming long lists that are freed. Breaking up the linked lists fixed the problem.</p>	7.0 SP3 8.1 SP1
CR102706	<p>In a cluster, a Managed Server attempting to update an Administration Server may have encountered a <code>java.lang.NullPointerException</code>.</p> <p>This problem occurred because of an issue with the registering of <code>NotificationListeners</code>.</p> <p>This problem was resolved by correcting the processing for registration/unregistration of <code>NotificationListeners</code> to protect against the possibility of a <code>NullPointerException</code> when attempting to unregister a <code>NotificationListener</code>, when no <code>NotificationListener</code> had been previously registered.</p>	8.1 SP1
CR102805	<p>The <code>weblogic.webservice.client.BaseWLSSLAdapter.addIdentity(java.security.cert.X509Certificate[] chain, java.security.PrivateKey privateKey)</code> method is no longer mistakenly marked as deprecated. The other two versions of this method are still correctly deprecated.</p>	8.1 SP1
CR102807	<p>A warning and a <code>java.lang.NullPointerException</code> may have occurred if a server name was not specified when first booting an Administration Server from a newly-created directory.</p> <p>This problem was resolved by modifying the code so that a management error is reported if a server name is not specified when booting an Administration Server.</p>	8.1 SP1
CR102849	<p>Domain log messages file resulted in a synchronous network IO, creating the possibility for deadlocks or other issues to arise.</p> <p>This problem was resolved by modifying the code so that messages are now broadcast to the domain log asynchronously.</p>	8.1 SP1
CR103117	<p>Customers who attempted to upgrade Web Service clients from WebLogic Server 7.1 to 8.1 may have experienced <code>ClassNotFoundException</code>s when the client was invoked using the <code>webserviceclient.jar</code>. This problem occurred because the <code>weblogic/webservice/PartFilter.class</code> and <code>weblogic/webservice/core/FaultMessage.class</code> files were missing from the <code>webserviceclient.jar</code>. They have now been added.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR103831	<p>A deadlock condition occurred when the server and domain logs were being rotated at the same instant on the Administration Server. (See also <a href="#">CR102849</a>.)</p> <p>This problem was resolved by modifying the code so that messages are now broadcast to the domain log asynchronously.</p>	8.1 SP1
CR103935	<p>After the error messages were printed to the domain log file, the server logging service was shut down. If the problem occurred, not all log messages were printed to the server log file. This problem occurred on both the Administration Server and the Managed Server(s).</p> <p>This problem occurred because in selecting the log file to rotate to, a log file that already existed on the file system would be chosen. Therefore, any attempt to rotate would result in a failure.</p> <p>When log files are rotating at a fast rate, it is possible that the file handle is still not released. As such, this problem was resolved by determining the oldest and latest files based on last modified timestamps, and offsetting the rotation size if any failure is encountered.</p>	8.1 SP1
CR104008	<p>The <code>weblogic.Admin SHUTDOWN</code> and <code>FORCESHUTDOWN</code> commands can now retrieve user credentials from a boot identity file. This enables you to invoke these commands from a script without storing unencrypted user credentials in the script.</p> <p>For more information, refer to <code>SHUTDOWN</code> and <code>FORCESHUTDOWN</code> in the <a href="#">WebLogic Server Command Reference</a>.</p>	8.1 SP1
CR104966 CR105251 CR105252	<p>The HTTP log rotation time format was not consistent with the format indicated in the online help. If the format specified in the online help was used, a parse error would be seen the next time the WebLogic Server instance was started, and it would not start.</p> <p>If the expected date format (<code>MM-dd-yyyy-k:mm:ss</code>) is not specified, the Administration Console now indicates the problem. The online help has been corrected to indicate this format.</p>	8.1 SP1
CR105198	<p>The following exception was displayed during startup of cluster instances after an Open LDAP Authentication provider was added to a default (active) security realm:</p> <pre>com.rsa.jsafe.JSAFE_PaddingException: Could not perform unpadding: invalid pad byte</pre> <p>This problem was resolved by having the Managed Server obtain the Administration Server's Encrypted Service and perform the proper Encryption.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR105777	<p>Exception propagation was not working properly because <code>RemoteException</code> was not being treated as a valuetype for rapid caching.</p> <p>Caching only real valuetypes fixed the problem.</p>	7.0 SP3 8.1 SP1
CR120295	<p>Some logging methods were inadvertently publicized.</p> <p>In the WebLogic Server 8.1 API Reference (Javadoc) for the <code>weblogic.logging.WLLogRecord</code> MBean, the setter methods should not have been made public. The attribute values in this MBean are precomputed and set internally; they should not be modified.</p> <p>In addition, in the API documentation for the <code>weblogic.logging.WLLevel</code> MBean, the constructor should not have been made public. The WebLogic logging services support a fixed set of levels, so creating your own <code>WLLevel</code> is not supported. You can extend the <code>java.util.logging.Level</code> class which is handled after appropriate mapping to <code>WLLevel</code> by the WebLogic logging service.</p>	8.1 SP1

## Tools

Change Request Number	Description	Release Fixed
CR069568	<p>The EJB 2.0 DTD allows for multiple roles and multiple methods to be defined inside one method-permission element. While WebLogic Builder allowed users to define multiple roles, it only allowed one method.</p> <p>This problem was resolved by modifying WebLogic Builder so that users can specify both multiple roles and multiple methods in one method-permission element.</p>	8.1 SP1
CR093246	<p>A new tag called <code>&lt;allow-remove-during-transaction&gt;</code> was added to the <code>weblogic-ejb-jar.xml</code> deployment descriptor. This tag was added for stateful session beans and is the child of the <code>&lt;stateful-session-descriptor&gt;</code> tag. However, the tag was not surfaced in WebLogic Builder.</p> <p>This problem was resolved by adding the 'Allow remove during transaction' check box to the Advanced Pane of a Session in WebLogic Builder.</p>	7.0 SP3 8.1 SP1
CR096719	<p>WebLogic Builder did not expose pool settings for entity beans.</p> <p>This problem was resolved by adding a Pool Panel for entity beans to the Tuning sub-node.</p>	7.0 SP3 8.1 SP1
CR099865	<p>Setting a default transaction value (<code>default-tx</code>) on a single bean using the <code>&lt;container-transaction&gt;</code> tag would result in the default transaction value being set the same way for all the beans in the same JAR.</p> <p>This problem was resolved by modifying the code so that setting the default transaction value on a single bean did not affect the same setting for other beans in the JAR.</p>	7.0 SP3 8.1 SP1
CR099913	<p>The automatic key generation frame in WebLogic Builder did not enable/disable all fields consistently. For example, if you opened a JAR that did not have an <code>&lt;automatic-key-generation&gt;</code> tag and loaded the Automatic Key Generation tab, the check box would be unchecked, but you could still enter a value.</p> <p>This problem was resolved by modifying the code so that the fields are uniformly disabled.</p>	7.0 SP3 8.1 SP1
CR102990	<p>WebLogic Builder no longer allows users to delete a role if it is being used in a method permission.</p>	8.1 SP1

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR103200	<p>If a Java source file that did not end in the <code>.java</code> extension was provided to <code>source2wsdd</code>, it would fail with the following error:</p> <pre>[source2wsdd] source2wsdd: Illegal package name</pre> <p>This was problematic since the convention for EJBGen source code files that are in split directories is for the suffix to be <code>.ejb</code>.</p> <p>This problem was resolved by modifying <code>source2wsdd</code> so that it would accept either a <code>.java</code> or <code>.ejb</code> suffix.</p>	8.1 SP1
CR105436	<p><code>java weblogic.marathon.ddinit.WebInit stageDir</code> was failing to generate servlet components in <code>weblogic.xml</code> and <code>web.xml</code>.</p> <p>A regression in the codeline cause the problem. This was fixed and the problem was resolved.</p>	7.0 SP3 8.1 SP1
CR108254	<p><code>wlcompile</code> failed when calling EJBGen if the classpath contained spaces.</p> <p>This problem occurred because when <code>wlcompile</code> forked EJBGen, it did not quote the classpath. As a result, if the classpath contained spaces, some of the elements were misinterpreted.</p> <p>This problem was resolved by making sure the classpath is treated correctly.</p>	8.1 SP1

## Utilities

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR101046	<p><code>weblogic.BuildXMLGen</code> overwrote an existing <code>build.xml</code> file without warning.</p> <p>This problem was resolved by adding the following warning message:</p> <pre>weblogic.utils.compiler.ToolFailureException: Build File already exists, use -file to specify alternative build file name.</pre>	8.1 SP1

## WebLogic Workshop

---

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR102094	<p>The script used to start the WebLogic Workshop sample server, <i>WL_HOME</i>\samples\workshop\startWebLogic.cmd was not installed in the WebLogic Server 8.1 release. Therefore, you could use the Windows Start Menu selection to start the sample server because it referred to this missing file.</p> <p>This problem has been fixed.</p>	8.1 SP1

---

# Web Services

Change Request Number	Description	Release Fixed
CR085000	<p>A <code>StackOverflowError</code> occurred during the serialization of a <code>java.lang.Object</code>. This problem was resolved by using <code>Object.toString()</code> for instances of the class <code>java.lang.Object</code>.</p>	8.1 SP1
CR087303	<p>When processing the following WSDL file, the clientgen ant task created invalid classes:</p> <pre data-bbox="292 656 1076 1135"> wsdl:message name="UploadCBProfileRequest"&gt; &lt;wsdl:part name="body" element="tns:TRAMSDATA"/&gt; &lt;wsdl:part name="authHeader" element="tns:AuthenticationHeader"/&gt; &lt;/wsdl:message&gt; &lt;wsdl:portType name="RelMgrPortType"&gt; &lt;wsdl:operation name="uploadCBProfile"&gt; &lt;wsdl:input message="tns:UploadCBProfileRequest"/&gt; &lt;/wsdl:operation&gt; &lt;/wsdl:portType&gt; &lt;wsdl:binding name="RelMgrSOAPBinding" type="tns:RelMgrPortType"&gt; &lt;soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/&gt; &lt;wsdl:operation name="uploadCBProfile"&gt; &lt;soap:operation soapAction="" style="document"/&gt; &lt;wsdl:input&gt; &lt;soap:body parts="body" use="literal"/&gt; &lt;soap:header wsdl:required="true" message="tns:UploadCBProfileRequest" part="authHeader" use="literal" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" /&gt; &lt;/soap:header&gt; &lt;/wsdl:input&gt; &lt;/wsdl:operation&gt; &lt;/wsdl:binding&gt; </pre> <p>The resultant generated task contained duplicate <code>uploadCBProfile()</code> arguments. Research revealed that the code was adding header parts to message even if that part was already there. A code fix resolved the problem.</p>	7.0 SP3 8.1 SP1
CR094458	<p>It is now possible to configure Web Service security requirements on a per operation basis for both requests and responses.</p>	8.1 SP1
CR096558	<p>This version of WebLogic Server uses a newer version of SOAP 1.2 specification (W3C Candidate Recommendation 19 December 2002).</p>	8.1 SP1

## Resolved Problems for Service Pack 1

Change Request Number	Description	Release Fixed
CR097887	<p>Web Service deployment descriptors and WDSL did not support all XML encodings. This may have resulted in the following error when using servicegen:</p> <pre data-bbox="229 491 1045 539">[servicegen] java.io.CharConversionException: Unconvertible UTF-8 character</pre> <p>This problem was resolved by using the <code>weblogic.webservice.i18n.charset</code> property to support encodings for Web Service deployment descriptors and WSDL.</p>	8.1 SP1
CR098050	<p>When deserializing a <code>xsd:string</code> to a <code>java.lang.String</code>, certain characters, for example <code>'/</code>, were causing the resulting String to be truncated .</p> <p>This problem was resolved by modifying the String deserializer to handle multiple character data events.</p>	8.1 SP1
CR099028	<p>A soap header with a simple type used the wrong namespace (schema namespace).</p> <p>This problem was resolved by fixing the soap header to use the target namespace of the Web Service.</p>	8.1 SP1
CR099030	<p>A one-way call to a WebLogic Server instance used to wait until the end component invoke was complete. Now the client returns after sending the request.</p>	8.1 SP1
CR100386	<p><code>javax.xml.rpc.Call.SESSION_MAINTAIN_PROPERTY</code> was not supported. In other words, nothing was done with this property if it was set on the client, which violated section 13.7 (Session Management) of the JAX-RPC 1.1 specification.</p> <p>This problem was resolved by adding support for this property.</p>	8.1 SP1
CR100528	<p>If you defined two Web Services using the same <code>serviceName</code>s and <code>serviceURIs</code>, as in:</p> <pre data-bbox="229 1190 986 1239">service serviceName="SessionFoo" serviceURI="ejb20test" service serviceName="SessionFoo" serviceURI="ejb20test"</pre> <p>both nodes would be merged into one Web Service (with a merged offering of operations).</p> <p>However, if you defined two Web Services using different <code>serviceName</code>s and the same <code>serviceURIs</code>, as in:</p> <pre data-bbox="229 1364 986 1413">service serviceName="SessionFoo" serviceURI="ejb20test" service serviceName="SessionBar" serviceURI="ejb20test"</pre> <p>only one of the Web Service's operations would show up under <code>./ejb20test</code>.</p> <p>This problem was resolved by adding a check in <code>servicegen</code> to be sure that the same <code>serviceURI</code> is not used in more than one Web Service. If it is, you will now get a build error.</p>	8.1 SP1

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR100807	The default method for Web Service back end components was not being called, therefore, the server-side dispatcher was not working properly.  This problem was resolved with a code fix.	8.1 SP1
CR100932	Default type mappings are now able to load the <code>types.xml</code> file from the classpath.	8.1 SP1
CR101149	When JMS transport was enabled and a browser was used to test a Web Service with multiple operations using different transports, the test page would display all operations in one section and did not distinguish between them.  This problem was resolved by fixing the test page so that it is now possible to distinguish between operations using different transports.	8.1 SP1
CR101160	In WSDL, it is possible to describe SOAP faults than contain complex types. This was not supported in the WebLogic container.  This problem was resolved by implementing sections 5.5.5 and 4.3.6 of the JAX-RPC specification.	8.1 SP1
CR101369	The <code>SOAPFaultException</code> <code>faultactor</code> and <code>faultstring</code> are no longer reversed.	8.1 SP1
CR101416	<code>Javax.xml.rpc.Service.getProxy( &lt;stub-class&gt; )</code> did not work correctly when one XML type was mapped to many Java types. This is fixed by using both Java and XML types to lookup for codecs.	8.1 SP1
CR102139	Web Services security now includes timestamps on signed messages. The security portion of the Web Service deployment descriptor has been extended to allow timestamp processing to be configured.	8.1 SP1

Change Request Number	Description	Release Fixed
CR102328	<p>A <code>ClassNotFoundException</code> was thrown while attempting to build a Web Service. While building the Web Service using the servicegen utility, an error starting with the following lines would occur:</p> <pre>java.lang.ClassNotFoundException: weblogic.management.descriptors.WebElementMBean at java.net.URLClassLoader\$1.run(URLClassLoader.java:195) at java.security.AccessController.doPrivileged(Native Method) at java.net.URLClassLoader.findClass(URLClassLoader.java:183) at java.lang.ClassLoader.loadClass(ClassLoader.java:294) at sun.misc.Launcher\$AppClassLoader.loadClass(Launcher.java:281) ) at java.lang.ClassLoader.loadClass(ClassLoader.java:250) ... </pre> <p>This problem occurred because WebLogic Server was not able to locate the <code>weblogic.jar</code> and <code>webservices.jar</code> files when they were not added to the system's <code>CLASSPATH</code>.</p> <p>This problem was resolved by modifying the code so that WebLogic Server always loads classes from the thread's context classloader.</p>	8.1 SP1
CR102686	<p>The asynchronous Web Service client failed and printed the following debug information when only the <code>webserviceclient.jar</code> was in the client's classpath:</p> <pre>java.lang.ClassNotFoundException: weblogic.kernel.Kernel </pre> <p><b>Note:</b> This problem was not encountered if <code>weblogic.jar</code> was also in the classpath.</p> <p>This problem was resolved by preventing the debug information from being printed, and by including <code>weblogic.kernel.Kernel</code> in the <code>webserviceclient.jar</code>.</p>	8.1 SP1
CR102772	<p>Adding a SOAP Header block with child elements, in a client-side handler without using a namespace prefix: (via <code>SOAPEnvelope.createName(String localName)</code> and <code>SOAPHeaderElement.addNamespaceDeclaration("","http://identifier.company.com/Context/v1";)</code> ) caused the <code>SOAPHeader.getChildElements(javax.xml.soap.Name)</code> method to return an empty <code>Iterator</code> against that SOAP Header block inside a server-side handler.</p> <p>This problem was resolved so that you can now add default namespaces by using <code>SOAPHeaderElement.addNamespaceDeclaration("","http://foo.bar/")</code>.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR102745	<p>The <code>weblogic.webservice.binding.jms.ConnectionPool.getInstance().close()</code> method, which is required only for light clients using JMS transport because of a bug in the SDK, has been added to the Javadocs.</p>	8.1 SP1
CR102985	<p>If a reliable request was made to a Web Service that did not exist, a SOAP fault exception was returned. The behavior of the sender was not deterministic in this case.</p> <p>This problem was resolved by modifying the code so that</p> <ol style="list-style-type: none"> <li>1. If there is a SOAP fault, a delivery failure is reported.</li> <li>2. If there are no reliability headers in the response, the request is retried.</li> <li>3. If the status is DUP or OK, the message is removed from the queue.</li> </ol>	8.1 SP1
CR103111	<p>A SOAP message was always treated as platform default encoding when it was encrypted and decrypted.</p> <p>This problem was resolved by allowing SOAP messages to have a specified encoding in the handler of the secure message. Web Service applications and clients that use WS-Security can now exchange non-ascii characters by specifying an encoding. The method of specifying an encoding is the same as for a non-secure SOAP message, except the <code>setAcceptCharset()</code> method in <code>weblogic.webservice.binding.BindingInfo</code> is ignored. If no encoding specified, UTF-8 is used.</p>	8.1 SP1
CR103394	<p>Optional parameters could not be set to <code>null</code> from a Java client using the stub generated from WebLogic Workshop. The optional parameters were always sent with default values.</p> <p>This problem was resolved by modifying the code so that optional parameters are not written or sent in the SOAP request from the client to the Web Service unless the value of the optional parameter(s) in the client is set in the client that uses the generated stub.</p>	8.1 SP1
CR103791	<p><code>java weblogic.webservice.autotype</code> did not handle arrays of user types as a Java type. For example, the first statement shown below would work, but the second would not:</p> <pre>java weblogic.webservice.autotype -destDir /tmp/z88 -javaTypes 'foo.bar.tests.SomeBean'</pre> <pre>java weblogic.webservice.autotype -destDir /tmp/z88 -javaTypes '[Lfoo.bar.tests.SomeBean;'</pre> <p>This problem was fixed by correctly loading Array classes.</p>	8.1 SP1

Change Request Number	Description	Release Fixed
CR104187	A serialization problem occurred when sending a SOAP message to a Web Service generated using <code>wsdl2service</code> from <code>GenericDataService.wsdl</code> .  This problem was resolved by modifying the code so that nested model types with <code>maxOccurs</code> set to a value greater than 1 work properly.	8.1 SP1
CR104719	Implementing a Web service with the following signature: <code>public void echoDom(Document doc)</code> resulted in a compile error.  This problem was resolved by disallowing void to be added to type-mapping for document style at build or runtime, preventing the writing of void part for document style, and avoiding using the wrapper element in the Document serializer for document style.	7.0 SP3 8.1 SP1
CR104989	<code>java weblogic.webservice.autotype</code> failed to generate serialization/deserialization classes for complex types that contained only references to named model groups.  This problem was resolved so that the classes are generated.	8.1 SP1
CR105353	Adding <code>useSOAP12="true"</code> to the <code>web-services.xml</code> file of a Web Service caused a <code>javax.management.InstanceAlreadyExistsException</code> to be generated during deployment of the Web Service.  This problem was resolved with a code fix.	8.1 SP1

## XML

Change Request Number	Description	Release Fixed
CR095744	The <code>XMLCacheRuntimeMBean</code> and <code>XMLCacheCumulativeRuntimeMBean</code> interfaces, which were not being used, have been deprecated and are now excluded from the Javadoc.	8.1 SP1
CR100798	In XMLDsig signed messages, the syntax for <code>X509SerialNumber</code> was invalid.  This problem occurred because <code>X509SerialNumber</code> was base64 encoded instead of just a String, and the problem was resolved by making this modification.	8.1 SP1

Change Request Number	Description	Release Fixed
CR102397	<p>The Xerces parser always attempted a call to <code>System.getProperty()</code>. This would throw an exception in an Applet, which does not have permission to retrieve properties. The following partial exception could be observed in an Applet making a JMS call:</p> <pre data-bbox="327 548 1096 826"> ----- Linked Exception ----- java.rmi.MarshalException: failed to marshal connectionCreate(Lweblogic.jms.dispatcher.DispatcherWrapper); nested exception is: java.rmi.UnexpectedException: Failed to parse descriptor file; nested exception is: java.security.AccessControlException: access denied (java.util.PropertyPermission weblogic.apache.xerces.maxentityrefs read) [...] </pre> <p>The code was modified so that the parser does not attempt to read system properties with Applet clients.</p>	6.1 SP6 8.1 SP1
CR103083	<p>Lookups on <code>xsd:include</code>'d types did not work for backward references. This problem was resolved by modifying the code so that if necessary, lookups properly went back up the include chain.</p>	8.1 SP1
CR103109	<p>For user-defined strings of simple type, the autotyper failed. The resulting generated code contained unqualified references to <code>java.lang.String</code>. This problem was resolved by making every usage of any class use the full package name.</p>	8.1 SP1

## Resolved Problems for Service Pack 1

# Resolved Problems for Service Pack 3

Service Packs are cumulative; Service Pack 3 contains all the fixes made in earlier Service Packs released for WebLogic Server 8.1.

The following sections describe problems that were resolved in WebLogic Server 8.1 Service Pack 3:

- [“Administration Console” on page 7-3](#)
- [“Classloaders” on page 7-5](#)
- [“Clusters” on page 7-7](#)
- [“Compilers” on page 7-10](#)
- [“Configuration Wizard” on page 7-10](#)
- [“Connector” on page 7-12](#)
- [“Core WebLogic Server” on page 7-14](#)
- [“Deployment” on page 7-22](#)
- [“EJB” on page 7-26](#)
- [“Installation” on page 7-32](#)
- [“J2EE” on page 7-33](#)
- [“JCOM” on page 7-34](#)
- [“JDBC” on page 7-35](#)

## Resolved Problems for Service Pack 3

- “jDriver” on page 7-41
- “JMS” on page 7-43
- “JNDI” on page 7-49
- “JSP” on page 7-49
- “JTA” on page 7-53
- “JVM” on page 7-55
- “Node Manager” on page 7-56
- “Operations, Administration, and Management” on page 7-56
- “Plug-Ins” on page 7-65
- “RMI/RMI-IIOP” on page 7-74
- “Samples” on page 7-77
- “Security” on page 7-77
- “Servlets” on page 7-86
- “SNMP” on page 7-96
- “Tools” on page 7-96
- “Web Services” on page 7-98
- “WebLogic Type 4 JDBC Drivers” on page 7-108
- “WebLogic Tuxedo Connector” on page 7-111
- “WLEC” on page 7-113
- “XML” on page 7-113

# Administration Console

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR111884 CR182433	<p>In WebLogic Tuxedo Connector, for imported services the combination of the resource name, local domain name, and remote domain name must be unique. However, the Administration Console prevented defining imported services with the same resource name.</p> <p>The Administration Console now enforces unique naming based on the combination of the resource name, local domain name, and remote domain name.</p>	8.1 SP3
CR125041	<p>A Connection Pool Waiter statistics monitoring area in the Administration Console sometimes showed an incorrect value.</p> <p>A check was added to ensure that the number of waiters is reported correctly.</p>	8.1 SP3
CR126506	<p>Users could not configure a console extension to replace the policy definition pages.</p> <p>Code was added to enable <code>CreateResourcesAction</code> to look for an extension before forwarding.</p>	7.0 SP5 8.1 SP3
CR127617 CR134432	<p>If a user did not have the correct permission to read files under a particular directory, <code>NullPointerException</code>s occurred in the console. Because the user did not have the correct permissions, empty arrays were returned.</p> <p>This problem was fixed by adding suitable checks. WebLogic Server does not display the files for which the user does not have permission.</p>	8.1 SP3
CR132229	<p>Console help was updated to properly identify a comma as the address separator when configuring Jolt and WLEC connection pools.</p>	8.1 SP3
CR132627	<p>When WebLogic Server started with a large heap size (such as <code>"-Xms3069m -Xmx3069m"</code> or <code>"-Xms2500m -Xmx2500m"</code>), the Administration Console displayed incorrect values under <code>server-&gt;monitoring-&gt;performance</code>. The maximum memory was always 1, and the current heap size was negative.</p> <p>This problem was corrected with a code fix.</p>	8.1 SP3
CR134434	<p>Symbolic links were resolved during the deployment process and absolute paths were saved in the configuration field (<code>config.xml</code>).</p> <p>A code change ensures that symbolic links are no longer resolved.</p>	8.1 SP3

Resolved Problems for Service Pack 3

Change Request Number	Description	Release Fixed
CR135294	<p>The console was not able to generate information from custom security MBeans used to create a custom security provider.</p> <p>A code fix was implemented so the console generated the related pages without any exceptions.</p>	8.1 SP3
CR137213 CR174743	<p>The <code>config.xml</code> was incorrectly updated on UNIX systems when the Administration Console was used to change the load order of Web applications. The leading '/' character in an application's path got lost.</p> <p>The code was fixed to ensure the attribute values are updated whenever they are changed from the related Console pages and the path attribute is not updated.</p>	8.1 SP3
CR158522	<p>Console help was updated to properly identify the range for polling interval when configuring the String Monitor, Gauge Monitor, or Counter Monitor. The correct range of values is between 1 and 65535 seconds.</p>	8.1 SP3
CR172232 CR178004	<p>When a connection pool using a custom JDBC driver was created, it was not possible to edit the advanced properties tab of the JDBC driver, and the server instance would throw a <code>java.lang.ClassNotFoundException</code> when the driver was not in the system classpath.</p> <p>A code fix was implemented to load the JDBC driver even if it is not found in the system classpath. The console displays the Connections--&gt;Advanced page without the XA attributes.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR173345	<p>The Administration Console now includes the following language preference settings:</p> <ul style="list-style-type: none"> <li>• Chinese Simplified/GB18030</li> <li>• Chinese Simplified/GB2312</li> <li>• Chinese Simplified/GBK</li> <li>• Chinese Simplified/UTF-8</li> <li>• Chinese Traditional/Big5</li> <li>• Chinese Traditional/Big5-HKSCS</li> <li>• Chinese Traditional/UTF-8</li> <li>• English</li> <li>• English/UTF-8</li> <li>• Japanese/EUC-JP</li> <li>• Japanese/Shift_JIS</li> <li>• Japanese/UTF-8</li> <li>• Korean/EUC-KR</li> <li>• Korean/UTF-8</li> </ul>	8.1 SP3

## Classloaders

Change Request Number	Description	Release Fixed
CR124348	<p>Client programs could not use <code>java.lang.reflect.Proxy</code> to access proxy objects deployed in WebLogic Server, unless the proxy classes were added to the system classpath. If an object did not reside in the system classpath, the client would receive a <code>ClassNotFoundException</code> exception.</p> <p>The <code>resolveProxyClass()</code> method was implemented to load interfaces from the application-specific classloader as well as the system classloader.</p>	<p>6.1 SP6</p> <p>7.0 SP5</p> <p>8.1 SP3</p>

Change Request Number	Description	Release Fixed
CR128510	<p>If a WebLogic Server 8.1 server returned a Calendar object to a WebLogic Server 6.1 or 7.0 client, the client got a <code>ClassNotFoundException</code>. This problem was caused by a change in the <code>Calendar</code> class in JDK 1.4. The <code>readClassDescriptor()</code> of <code>MsgAbbrevInputStream</code> tried to resolve the class, leading to a <code>ClassNotFoundException</code> for unknown classes. Java serialization skipped this <code>ClassNotFoundException</code> if corresponding data was not being read.</p> <p>The <code>MsgAbbrevInputStream.readClassDescriptor()</code> no longer tries to resolve the class, and <code>MsgAbbrevInputStream</code> now implements <code>resolveClass()</code>.</p>	7.0 SP5 8.1 SP3
CR137120	<p>The following error was encountered when sequentially starting two Managed Servers in the cluster. The error was not encountered if the two Managed Servers were started simultaneously.</p> <p><code>LinkageError : duplicate class definition</code></p> <p>The problem was resolved with a code change.</p>	8.1 SP3
CR161884 CR173695	<p>EAR files gave "<code>NoClassDefFoundError</code>" and "<code>IllegalStateException: zip file closed</code>" messages on classes that actually exist in the WAR file.</p> <p>When multiple threads tried to load a class from a WAR file within an EAR file, it was possible that some JAR files within the <code>WEB-INF/lib</code> directory were not looked into; resulting in <code>NoClassDefFoundError</code> and <code>IllegalStateException</code>.</p> <p>This problem was resolved with a code change, which added necessary synchronization to ensure that the zips structure within the Finders are not corrupted.</p>	8.1 SP3

# Clusters

Change Request Number	Description	Release Fixed
CR111029	<p>The cluster members timed out if they did not receive a heartbeat within the default 30 second timeout period. Heartbeats were sent every 10 seconds and servers waited for 3 periods (total wait time was 30 seconds) to get a heartbeat before the cluster member was timed out and declared unavailable. For example, during session replication if the secondary server was unavailable at TCP level, the 30-second period was sometimes too long for a very busy web site. Before the secondary was removed from the cluster view, the primary tried to replicate many sessions to the secondary and thus caused the server to hang or made the server slower.</p> <p>The timeout value <code>IdlePeriodsUntilTimeout</code> is now tunable. It is set on the <code>&lt;Server IdlePeriodsUntilTimeout="3"&gt;</code> tag in the <code>config.xml</code> file. In general customers should not tune this value and should leave it at the default (3). However, in certain cases depending on the load, available redundancy in the architecture and specific application problems and/or certain production scenarios, tuning this value carefully might alleviate the problem temporarily until the root cause is identified and fixed.</p> <p>BEA WebLogic recommends that you use HIGH caution when changing this value and ensure sufficient testing of your application at peak load scenarios to ensure the expected behavior. There is no recommendation that fits all scenarios, so testing for load and stress is a must if this value needs to be changed.</p>	8.1 SP3
CR112326	<p>A memory leak occurred with <code>weblogic.cluster.BasicServiceOffer</code> during JNDI rebinds of replicable objects. The problem was resolved with a code correction.</p>	7.0 SP5 8.1 SP3
CR112874	<p>When a client of a stateful session bean accessed a bean deployed on a cluster configured for weight-based load-balancing, and the Managed Servers with the highest and second highest weight were killed in that order, the client gave the following message:</p> <pre>&lt;Jul 22, 2003 1:32:56 AM IST&gt; &lt;Warning&gt; &lt;Kernel&gt; &lt;No replica in list has the expected weight. Reverting to round-robin&gt;</pre> <p>When the Managed Servers were restarted, the load balancing algorithm switched to round-robin. Analysis revealed that the replica list was updated when a Managed Server went down, but due to a race condition the max weight in <code>RichReplicaList</code> was not reset properly. A code change to recompute max weight whenever the replica list size changes solved the problem.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR121113	<p><code>HttpClusterServlet</code> did not print thread name in error messages, making it difficult to diagnose the problem when an error occurred.</p> <p>A code change was made to <code>GenericProxyServlet.trace()</code> to add the thread name in logged error messages to aid in troubleshooting.</p>	7.0 SP5 8.1 SP3
CR122924	<p>When cluster address was specified as <code>t3://127.0.0.11:8001,127.0.0.12:9001</code></p> <p>WebLogic Server incorrectly appended the port number when generating the home handles.</p> <p>WebLogic Server now confirms whether the port number is already specified by the user and uses that information to properly create the URL used by the home handle.</p>	8.1 SP3
CR126406	<p>The cluster service set the real uid/gid of the process after binding to the multicast port, which caused errors when binding the server to listen ports.</p> <p>A code change was made to ensure that the server only uses privileged uid/gid when binding to listen ports. For the rest of the server startup, the unprivileged uid/gid is used.</p>	8.1 SP3
CR127375	<p>In a single node cluster using a weight-based load balancing algorithm, a remote method invocation on an EJB from an external client resulted in an <code>ArrayIndexOutOfBoundsException</code>.</p> <p>When there is a single node cluster, WebLogic Server will not have any replicas for requests to fail over or load balance. Adding a check for this condition resolves the problem.</p>	8.1 SP3
CR127616 CR133899	<p>A <code>NullPointerException</code> sometimes occurred in the <code>BasicReplicaHandler</code> during 70-&gt;81 interop if cluster affinity is set.</p> <p>A null check was added to all affinity handlers.</p>	8.1 SP3
CR130407	<p>A client attempted to fail over even when there was only one server in the cluster.</p> <p>This was resolved with a code fix so that failover happens only when there is more than one server in the cluster.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR130507	<p>Session replication caused server nodes to hang in a cluster because batch updates were performed for the <code>lastAccessedtime</code> of the sessions, which were all stored in a hashtable for all sessions. While doing batched updates, this hashtable was locked until all the updates were done (all the updates were remote calls), thus preventing other threads from updating the hashtable and resulting in poor performance.</p> <p>The code was fixed so that a new hashtable (local variable) is created inside the trigger method of <code>BatchedLATUpdater</code>, which is a clone of the original hashtable. The new hashtable is used to access the <code>lastAccessstime</code> for the sessions; the old hashtable is synchronized only while it is being cloned. Therefore, it can be used by other threads while the remote calls are being made.</p>	8.1 SP3
CR177367	<p>An NPE was thrown when running a cluster under a heavy load or when the session stickiness was not maintained. When the load was high on the cluster nodes the plug-in was getting a connection refused from the node and hence failing over the requests to the other nodes in the cluster. Either condition was causing the session to change/get created randomly on any node in the cluster, and in certain scenarios it was possible that the primary was removed and a secondary was created for the session on the same node. Any thread trying to get the secondary information was failing with an NPE because finding the secondary information requires the primary. While the primary was removed first before creating secondary, the <code>secondaryURL</code> was being computed to <i>null</i>.</p> <p>A code fix ensure that WebLogic Server now checks for <i>null</i> <code>secondaryURL</code>. In addition, a warning message is now logged to inform users of the problem, so they can configure the cluster for the expected load. This also ensures that the situation can be debugged if the session stickiness is broken in the load balancer configuration or the plug-in.</p>	8.1 SP3

## Compilers

Change Request Number	Description	Release Fixed
CR136199	<p>EjbObject.idl generated by ejbc -idl was not compliant with CORBA3.0.</p> <p>From CORBA 2.6: "3.2.3.1 Escaped Identifiers. As IDL evolves, new keywords that are added to the IDL language may inadvertently collide with identifiers used in existing IDL and programs that use that IDL. Fixing these collisions will require not only the IDL to be modified, but programming language code that depends upon that IDL will have to change as well. The language mapping rules for the renamed IDL identifiers will cause the mapped identifier names (e.g., method names) to be changed. To minimize the amount of work, users may lexically "escape" identifiers by prepending an underscore ( <code>_</code> ) to an identifier. This is a purely lexical convention that ONLY turns off keyword checking. The resulting identifier follows all the other rules for identifier processing. For example, the identifier <code>_AnIdentifier</code> is treated as if it were <code>AnIdentifier</code>."</p> <p>The problem was resolved by a code change that adds an underscore ( <code>_</code> ) to known IDL reserved words.</p>	8.1 SP3

## Configuration Wizard

Change Request Number	Description	Release Fixed
CR128745	<p>When using the Domain Configuration Wizard in silent mode to create a domain configuration, the Administration Console showed a user assigned to same group more than once.</p> <p>The code was modified so that it does not add an entry if the corresponding assignment is already done.</p>	8.1 SP3
CR130268	<p>When the Domain Configuration Wizard encountered a problem that resulted in a failure it was returning a zero.</p> <p>The code was modified to return a non-zero when user script files are not found.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR130536	<p>On the AIX 5.1 platform running WebLogic Server 8.1, Service Pack 1 with the Posix Performance Pack, a SIGSEGV 11 (*) segmentation violation occurred from the IBM JVM.</p> <p>To recreate this error:</p> <ol style="list-style-type: none"> <li>1. Create a new WebLogic domain using the Configuration Wizard.</li> <li>2. Set the file descriptor limit to <i>unlimited</i> (<code>ulimit -n</code>) in the shell.</li> <li>3. Run the <code>startWebLogic.sh</code> script.</li> </ol> <p>This issue was resolved with a code fix that ensures that the <code>resetFd()</code> method is called in the <code>commEnv.sh</code> script, which automatically resets the <i>unlimited</i> setting to 1025.</p>	8.1 SP3
CR132234	<p>Silent-mode script can now be used to create and edit the log child element of a Server element.</p> <p>Silent-mode configuration can be used to create and edit any valid configuration object and its child elements, except custom security objects. For more information about valid configuration objects, see the <a href="#">WebLogic Server Configuration Reference</a>. For more information about silent-mode script configuration, see “<a href="#">Creating a Script for Silent-Mode Configuration</a>” in <i>Creating WebLogic Configurations Using the Configuration Wizard</i>.</p>	8.1 SP3
CR132608	<p>While attempting to create a domain using WebLogic Server Template Builder on Solaris 2.9 and creating a MySQL JDBC connection pool, the JDBC URL failed.</p> <p>This problem was fixed with a code fix.</p>	8.1 SP3
CR135642	<p>When running the Configuration Wizard in silent mode, the wizard incorrectly converted the JDBC connection pool URL for the Oracle OCI driver (<code>jdbc:oracle:oci:@&lt;DBName&gt;</code>) to the URL for the Oracle Thin driver (<code>jdbc:oracle:thin:@&lt;DBHost:port:DBName&gt;</code>). This problem prevented deployment of JDBC connection pools with the Oracle OCI driver when the server was started.</p> <p>The Configuration Wizard now accepts the URL for either Oracle JDBC driver.</p>	8.1 SP3

# Connector

Change Request Number	Description	Release Fixed
CR124102	<p>When the process of enlisting an adapter's connection with a transaction threw an exception, the state of the connection was not properly reset, which meant the connection could never be cleaned up by the shrinker.</p> <p>Also, the connection could not be used by incoming application requests. Eventually, this resulted in exhaustion of the connection pool.</p> <p>The server code has been repaired to properly reset the state of the connection when the enlistment fails. The connection can be cleaned up by the shrinker, and can also be used by future application requests.</p>	7.0 SP5 8.1 SP3
CR125399	<p>For adapters that failed the connection proxy test, connection idle detection should have been disabled even if inactive-connection-timeout-seconds is specified in the weblogic-ra.xml descriptor. Prior to this release, the server tried to perform idle cleanup using such an adapter, which would result in a NullPointerException.</p> <p>Now idle-detection is properly disabled for an adapter that fails the proxy test. As a result, the NullPointerException will no longer occur.</p>	8.1 SP3
CR126184 CR135205	<p>In some Resource Adapters, the Connector container sometimes reported false connection leaks. The problem occurred for adapters which implement the ManagedConnection.getConnection() method in a way that the same handle could be returned in subsequent calls.</p> <p>The problem occurred because of the way it created proxies around the connection handles. When a proxy was finalized, and there was a handle with the same hashCode value still open, the container behaved as if there was a leak.</p> <p>The Connector container now keeps track of a count associated with handles that have the same hashCode. The count is incremented when opening and decremented when the proxy is finalized. In this way, the container will only report a leak if all the proxies associated with a handle are finalized, and no false leaks will be reported.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR129392 CR133572 CR177908	<p>A WebLogic Server instance hosting a Messaging Bridge had to be restarted when the MQSeries 5.3 Queue Manager was stopped and restarted. This was because the call into MQSeries would fail when the Messaging Bridge attempted to close an XASession object. This resulted in resources not being cleaned up with regards to the MQSeries Queue Manager after shutting it down. Also, the Queue Manager would not restart because it detected that there were still objects in use by the WebLogic Server process.</p> <p>Application of the IBM upgrade CSD06 to MQSeries 5.3 resolves this issue. As a result, the Queue Manager can be shutdown and restarted without error, and without having to shutdown the WLS server process.</p>	8.1 SP3
CR174339	<p>Some internally cached objects were not cleaned up properly when connections were permanently removed from the pool, which could then result in the cache growing arbitrarily large.</p> <p>A code change ensures that cached objects are cleaned up when a connection is permanently removed from the pool.</p>	8.1 SP3
CR175301	<p>Under a heavy load with the jRockit JVM, the <code>ConnectionRuntimeMBeanImpl()</code> API for WebLogic Connector sometimes throws an <code>InstanceAlreadyExists</code> exception. This is because the <code>ConnectionInfo#hashCode()</code> method is not always unique for different objects.</p> <p>In previous service packs, the hashcode of the connection was used to generate the MBean name. The code was improved so that it now uses a synchronized counter to generate the name. As a result, all runtime MBeans used for monitoring resource adapter connections are generated with unique names.</p>	8.1 SP3
CR176940	<p>When deploying an adapter on multiple servers in a cluster, the connections created on each server got an XAResource, and the server used the same name for resource registration. This caused the Transaction Manager to try to invoke calls on the wrong XAResource for some operations.</p> <p>The name used for resource registration is now qualified by the server and domain name, thus ensuring that an adapter deployed on different servers in a cluster will use different names for the resource registration.</p>	8.1 SP3

## Core WebLogic Server

Change Request Number	Description	Release Fixed
CR098607 CR136879	<p>In a situation in which Application A is using Application B, while looking up Application B, the J2EE container of Application A creates a <code>DependencyClassLoader</code> by attaching Application B's <code>ClassFinder</code> to it. On redeployment of Application B, Application B encounters a new <code>ClassFinder</code> and its old <code>ClassFinder</code> is no longer valid. On a client request, the server attempts to use the old <code>DependencyClassLoader</code> and throws an exception.</p> <p>The behavior that caused the exception has been changed: WebLogic Server no longer uses <code>DependencyClassLoader</code> for loading impl class from <code>ReplicaAwareRemoteObject.getPrimaryRepresentative()</code>.</p>	7.0 SP5 8.1 SP3
CR105257 CR174955	<p>If a call to <code>WLECSERVICE.getConnectionPoolCount</code> was made before the IIOP Connection Pool was initialized it resulted in a NPE.</p> <p>Code was added to return a 0 if the IIOPConnection Pool was null.</p>	8.1 SP3
CR105444	<p>The server would throw a <code>FileNotFoundException</code> if a path for a log file was specified but the directory not exist when using the <code>-Dweblogic.Stdout</code> and <code>-Dweblogic.Stderr</code> command-line options.</p> <p>A code enhancement resolves this issue by creating the directory structure specified by the command-line options when the structure does not exist.</p>	8.1 SP3
CR108993 CR133901	<p>On HP-UX11, the CPU usage was high when using the native posix muxer.</p> <p>A new muxer was designed, based on <code>/dev/poll</code>. This muxer can be enabled by setting the <code>MuxerClass</code> property as follows -</p> <pre>&lt;Server ... MuxerClass="weblogic.socket.DevPollSocketMuxer" ...&gt;</pre> <p>The following message should appear when the server starts up. This message confirms that the <code>/dev/poll</code> has been enabled.</p> <pre>&lt;Sep 9, 2003 5:29:28 PM PDT&gt; &lt;Info&gt; &lt;Socket&gt; &lt;BEA-000406&gt; &lt;DevPollSocketMuxer was built on Sep 8 2003 15:34:11&gt;</pre> <p>If the <code>/dev/poll</code> device is not available, it will fail to load. Currently it is available for:</p> <ul style="list-style-type: none"> <li>• Solaris 7 with a patch and on Solaris 8 and above</li> <li>• HP-UX 11.0 with PHKL_24064 patch required</li> <li>• HP-UX 11i with PHKL_25468 patch required</li> </ul>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR109180	<p>Transactions started on one server that used another server as the transaction coordinator would fail during the commit if the two servers have not previously exchanged transaction contexts. This occurred because the channel being passed to the JTA interceptor contained information on the local server, not the remote peer.</p> <p>A code fix was used to pass correct channel to the JTA interceptor.</p>	8.1 SP3
CR109307	<p>Previously you could not bring up a WebLogic Server 8.1 SP2 Managed Server in an 8.1 SP3 domain. A code change has brought the different service packs into compatibility and into accord with the WebLogic Server compatibility statement: "Servers within an Administrative domain can be at different Service Pack levels as long as the Administration Server is at the same Service Pack Level or higher than its Managed Servers."</p>	7.0 SP5 8.1 SP3
CR109688 CR133876 CR133877 CR135235 CR176217	<p>Under certain conditions a simple Java client looking up <code>InitialContext</code> would receive an incorrect "No version information" error.</p> <p>The problem was that line breaks in the header buffer were marked by "\n" instead of by the <code>line.separator</code> property. A code change corrected the problem.</p>	7.0 SP5 8.1 SP3
CR116707	<p>WebLogic Server provided a <code>ReplicaAwareRemoteRef</code> warning log when you used certain <code>Factory/Trader</code> patterns, displaying the following on the client side:</p> <pre>[ReplicaAwareRemoteRef] : WARNING: client-side RA stub didn't find environment on thread</pre> <p>Following a code change, the inappropriate log is no longer printed to the client.</p>	7.0 SP5 8.1 SP3
CR116712	<p>In WebLogic Server 6.1 SP04, the <code>WLECCConnectionRuntime</code> MBeans were not updated when an <code>INVOKE</code> was issued on <code>WLECCConnectionPoolRuntime</code> using Admin tool.</p> <p>Analysis revealed that when <code>resetConnectionPool()</code> was invoked on the <code>WLECCConnectionPoolRuntime</code> MBean, the pool was not reset. Old connections were not removed.</p> <p>The problem was solved with a code change. Now, the mbeans that correspond to old connections are unregistered before the <code>resetConnectionPool()</code> invocation.</p>	6.1 SP6 8.1 SP3
CR120492 CR122551	<p>To improve HTTP tunneling performance, a code change sets the content length for tunneled responses to maintain keep-alive connections, and also enables caching of HTTP URL connections.</p>	7.0 SP5 8.1 SP3
CR120492 CR122551 CR171857	<p>To improve HTTP tunneling performance, a code change has set the content length for tunneled responses to maintain keep-alive connections, and has also enabled caching HTTP URL connections.</p>	7.0 SP5 8.1 SP3

Change Request Number	Description	Release Fixed
CR120811	<p>When using the WebLogic thin client with an applet, concurrentModificationExceptions and JMSEExceptions were thrown. Investigation showed that there were two problems:</p> <ul style="list-style-type: none"> <li>• There was a problem with the Sun ORB implementation. An applet's virtual machine released AppletContext upon a browser refresh and stopped all threads in the applet context's thread group. When an ORB was initialized as part of an applet context, the reader threads were created in the applet context's thread group. When the browser was refreshed, the ORB reader threads were also stopped.</li> <li>• The WebLogic thin client created two threads in the applet context group: a HeartbeatMonitor thread and a RequestTimer thread. When the browser was refreshed, these threads were stopped with others in the applet context group.</li> </ul> <p>The problems were solved with the following changes:</p> <ul style="list-style-type: none"> <li>• The Sun ORB implementation changed in JDK 1.4.2_04 so that it creates the reader threads on a child thread group of the system thread group but not to the applet's context thread group. This change ensures the reader thread stays alive as long as the orb is alive or applet's JVM is alive.</li> <li>• The WebLogic thin client TunnelResponse and HeartbeatMonitor threads are now created on a child thread group of the system thread group but not to the applet's context thread group. This change ensures these threads stay alive as long as applet's JVM is alive. The fix is provided only for signed applets.</li> </ul>	8.1 SP3
CR121311	<p>Server shutdown failed because of in-use resources. This was due to application problems: either an application leaked pooled resources, such as JDBC connections and JMS sessions, or an application was holding a reference to pooled resources after server shutdown had commenced.</p> <p>By default, pools are now shut down even if resources are in use.</p> <p>For JDBC connections, to revert to the previous behavior, set ignoreInUseConnections="false" on the JDBCConnectionPoolMBean.</p>	8.1 SP3
CR122361	<p>The Auditing enabled message did not record the USER identity:</p> <pre data-bbox="229 1333 1045 1385">&lt;Aug 29, 2003 12:31:25 AM EDT&gt; &lt;Info&gt; &lt;Configuration Audit&gt; &lt;159909&gt; &lt;Configuration Auditing is enabled&gt;</pre> <p>This was inconsistent with the disable message, which did record the USER:</p> <pre data-bbox="229 1442 1045 1494">&lt;Aug 29, 2003 12:31:31 AM EDT&gt; &lt;Info&gt; &lt;Configuration Audit&gt; &lt;159910&gt; &lt;USER system, Configuration Auditing is disabled&gt;</pre> <p>Following a code change, both audit messages record the USER identity.</p>	7.0 SP5 8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR122706	<p>A clustered Web application deployed on a cluster that called an EJB on another cluster threw an <code>AssertionError</code>. Analysis revealed that generated code did not use the right method while calculating methods in the interfaces, which resulted in using the wrong classloader from the generated stub.</p> <p>A code change resolved the problem.</p>	7.0 SP5 8.1 SP3
CR122878	<p>MBeans were being deleted by <code>KernelID</code>. As a result, no matter who initiated the delete operation, the Auditor recorded that the Kernel had deleted the MBean.</p> <p>Following a code change, the MBean is deleted by the current Authenticated Subject.</p>	7.0 SP5 8.1 SP3
CR123509	<p>Performance degradation occurred when tunneling the t3s protocol through IPlanet for HTTP/1.0 requests because of an IPlanet change related to the Keep-Alive functionality. This problem was resolved by a code change that now specifies HTTP/1.1 for tunneling requests.</p>	7.0 SP5 8.1 SP3
CR123571 CR128843 CR183237	<p>When starting several T3 clients on the same machine simultaneously, there was a possibility that two or more of the clients could obtain the same JVMID, cause exceptions or hanging on the clients. The problem occurred only when starting multiple T3 clients on the same machine at the same time. This problem was solved by modifying the code used to generate JVMIDs.</p>	6.1 SP6 8.1 SP3
CR125245	<p>The <code>java.lang.LinkageError: duplicate class definition</code>: error sometimes occurred when multiple threads attempted to load the same class in a Web application while <code>PreferWebInfClasses</code> was enabled. This problem occurred the first time the classes were loaded.</p> <p>A code change to synchronize the <code>loadClass</code> method of <code>ChangeAwareClassLoader</code> resolved this problem.</p>	7.0 SP5 8.1 SP3
CR125285	<p>WebLogic Server was masking certain exceptions during application deployment because of an unhandled internal exception. With the code fix which handles this exception, the original exceptions are rethrown, which makes it easier to debug the original issue.</p>	7.0 SP5 8.1 SP3
CR125440	<p>WebLogic Server cached the <code>EndPoints</code> based on <code>ConnectionKey</code> for <code>iiop</code> requests. The <code>ConnectionKey</code> was based on host, port and type information on which the connection was established. Because of the way the <code>ConnectionKey</code> was established, tunneling requests used the wrong <code>Connection</code> object. Some clients from the same host never got responses back under load test because they were going to different <code>EndPoints</code>.</p> <p>WebLogic Server now creates unique IDs for a <code>Connection</code> object so that tunneling requests use the correct <code>Connection</code> object.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR126374	<p>WebLogic Server was not calling the <code>setPriority</code> method on an <code>ExecuteThread</code> even when a priority was set for a particular execute queue.</p> <p>Code was added to call <code>setPriority</code> when an <code>ExecuteThread</code> gets created.</p>	8.1 SP3
CR126613	<p>When <code>onMessage</code> for an MDB was invoked on a thread, the following exception was thrown because the transaction previously associated with the thread was not cleaned up properly:</p> <pre>weblogic.utils.AssertionError: ***** ASSERTION FAILED ***** [ Left-over JTA transaction found on MDB listener thread ]</pre> <p>A stale transaction on the thread is now cleaned up before the thread is used to call the <code>onMessage</code> method on an MDB.</p>	8.1 SP3
CR127394	<p>When <code>weblogic.jar</code> was added as a <code>ClassPathElement</code> inside the ant task , ant used <code>AntClassLoader</code> to load classes from the <code>weblogic.jar</code>. This process worked, but ant did not properly set the <code>contextclassloader</code> on the thread. The <code>context classloader</code> on the thread was still the <code>system classloader</code>. This resulted in classloading issues while generating the stub.</p> <p>This problem has been fixed in the <code>WLDeploy</code> ant task that WebLogic Server creates. In the implementation of the <code>WLDeploy</code> task the correct <code>context classloader</code> is set.</p>	8.1 SP3
CR127455	<p>The Administration Server uses a trigger to send a heartbeat to each Managed Server to verify that the Managed Server is running. When a Managed Server is restarted, the trigger for the old Managed Server needs to be cancelled. In some situations, the trigger for the old Managed Server was not cancelled properly.</p> <p>The code was fixed to ensure the trigger is correctly cancelled.</p>	8.1 SP3
CR128417 CR134918 CR172274	<p>Sometimes server-side sockets remained in an IDLE state when a client issued a reset on the socket before the server could set the <code>TCP_NODELAY</code> option. Therefore, these IDLE sockets were not getting cleaned up properly.</p> <p>A code change ensures the clean up of server-side sockets when an exception occurs when setting the <code>TCP_NODELAY</code> option on the socket.</p>	8.1 SP3
CR128445	<p>The <code>weblogic.Admin</code> command line utility set the username and password for the <code>ServerStartMbean</code> each time it was used to start a server instance. If a user attempted to start a server with <code>weblogic.Admin</code>, it would fail without permissions to WRITE attributes.</p> <p>A code change ensures that WRITE attributes not set on <code>ServerStart</code>.</p>	7.0 SP5 8.1 SP3

Change Request Number	Description	Release Fixed
CR128496	<p>Attempted interoperability between WebLogic Server 6.1 SP5 and 8.1 SP1 domains caused an <code>InvalidClassException</code>. This is because 6.1 SP5 was released with the pre-EJB 2.0 specification, while release 7.0 and later were released with the final EJB 2.0 specification, which changed the class <code>javax.ejb.TransactionRolledbackLocalException</code> in the final release. Therefore, if a release 6.1 SP5 domain sends this exception to an 8.1 domain, then the 8.1 domain fails to read the exception while deserializing this exception object with <code>InvalidClassException</code> because the SVUID of the local class and incoming response class are different.</p> <p>The code has been modified to load the local class and stamp the SVUID to the incoming class when the other peer is a release 6.1 domain.</p>	8.1 SP3
CR128646	<p>When running a servlet, the generic class loader could not find the WebLogic JDBC class.</p> <p>This was resolved by adding a check to the generic class loader to see if the JDBC class is set to <code>null</code> in <code>weblogic.utils.wrapper.WrapperFactory.java</code>. If it is set to <code>null</code>, then the current <code>ClassLoader(this.getClass().getClassLoader())</code> is loaded.</p>	8.1 SP3
CR129094	<p>Performance issues that involved TCP window shrinkage in the t3 protocol on the AIX platform have been resolved.</p>	7.0 SP5 8.1 SP3
CR130376 CR133631 CR174605 CR175607 CR172366	<p>According to the documentation in “<a href="#">Starting and Stopping Servers</a>” if the CLASSPATH is too long, it can be added as a single line to a file and then accessed as <code>-classpath @filename</code>. However, when <code>beasvc</code> attempted to load the contents of CLASSPATH file, it truncated the last character when the file did not end with a new line.</p> <p>Now <code>beasvc</code> determines whether the file terminates properly and then reads the file accordingly.</p>	8.1 SP3
CR130409 CR178422	<p>When setting the maximum length of an execute queue with the <code>-Dweblogic.kernel.allowQueueThrottling</code> flag to throttle “slow moving resource intensive” requests on a custom queue, clients did not receive a 503 response and therefore waited for the timeout.</p> <p>This problem was resolved by a code fix to check for the dispatch return value on the caller and use the <code>sendError()</code> API to return the 503 response.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR132704	<p>When using the <code>MBean.forceShutdown</code> from a servlet's <code>Init</code> method and using load-on-startup, the server did not successfully shut down and the "Start this Server" button in the Control tab is greyed out and stays in this state for quite some time. After several minutes, the link is enabled.</p> <p>The problem was resolved with a code change. The <code>forceShutdown</code> call now ensures the server is shut down without hanging the server lifecycle thread. Because the server state is properly changed, the console properly activates the links as soon as the server is down.</p>	8.1 SP3
CR132994	<p>Due to a Sun plug-in bug, a lookup for an EJB was failing. However, a WebLogic Server bug was exposed because of this issue. There was only one server in the cluster and yet the system was trying to fail over to another server even when the replica list was empty.</p> <p>This problem was resolved with a code change; failover only occurs if the replica list is not empty.</p>	8.1 SP3
CR135122	<p>When an application in a WebLogic server instance made an RMI call to another application within the same server instance, it uses the Call By Value (CBV) Wrapper implementation. The CBV wrapper implementation cloned the values for all the object instances except primitive datatypes and Date instances. If application specific code derived the Date and provided it's own implementation for the Date class, cloning the user defined class created an object instance on the other server's classloader. A <code>ClassCastException</code> occurs when the other server assigned the instance to the caller's application.</p> <p>The problem was resolved with a code change.</p>	8.1 SP3
CR135255	<p>When a request comes over a network channel, the server (instead of writing the default channel information as part of JVMID) writes the address information of the network channel. This information from streams is used to detect if the request came over a default channel or a network channel. In this case, the <code>ChunkedObjectOutputStream</code> did not implement the <code>EndPointStream</code> interface because the server was unable to make a distinction between the default and network channel.</p> <p>This problem caused a degradation in performance.</p> <p>A code fix ensures that the <code>ChunkedObjectOutputStream</code> implements the <code>EndPointStream</code> interface so that the JVMID class can determine on which channel the request came.</p>	8.1 SP3
CR135256	<p>The server wrote the address information of the default channel rather than the network channel information as part of JVMID. This caused the client to throw a <code>java.rmi.ConnectException</code> because the default channel was not visible to the client.</p> <p>A code fix ensures the server writes the address information correctly.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR136076	<p>EJB network channel issue; unable to connect to the business channel. When the connection request was received on the network channel, the server correctly translated the JVMID by writing the network channel information in it. However the server was also translating the rawAddress and calculating it based on the network channel address. The rawAddress must be calculated based on the defaultChannel address so that JVMID equality is maintained.</p> <p>A code fix was implemented to always write the rawAddress based on the defaultChannel address, even if the request is received on a network channel.</p>	8.1 SP3
CR136250	<p>When a request was retried for proxy authentication, the original headers and post data were not sent by HttpURLConnection, which caused Proxy Authentication to fail.</p> <p>A code fix ensures that any headers and data from the original request are re-sent during a retry.</p>	8.1 SP3
CR136756	<p>A client threw a <code>java.net.UnknownHostException</code> exception when it tried to connect to a server. The exception was thrown because the client had previously connected to a server that had a private network channel. The client threw the exception when it was unable to determine if a connection existed before attempting to open a new connection.</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3
CR172105 CR177136	<p>Calling the <code>setTimeout</code> method of <code>weblogic.net.http.HttpURLConnection</code> no longer takes twice the amount of time specified.</p>	8.1 SP3
CR172366	<p>Messages printed by <code>beasvc</code> to the event log were not readable in Japanese locale.</p> <p>A code fix ensures that English messages are printed for all non-English locales.</p>	8.1 SP3
CR173345	<p>When trying to determine if a client was an applet, WebLogic Server tried to read a system property, and if it threw a security exception the client was assumed to be an applet. For signed applets a security exception was never thrown and it was incorrectly assumed to be a Java client when it was actually an applet. Therefore, WebLogic Server could not distinguish between a Java client and a signed applet.</p> <p>A code change ensures that in addition to determining whether a client is a signed applet, WebLogic Server also reads additional properties that are only set for an applet.</p>	8.1 SP3
CR173958	<p>Interoperability between 6.1 SP4 and 8.1 SP2 domains using <code>tc3</code> failed when the protocol was changed from “secure” to “non-secure” between the front-end and back-end. The front-end QOS was being propagated to the back-end for the authentication call and it was failing.</p> <p>The problem was resolved by using “anonymous” when doing the bootstrap authenticate call.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR174605	<p>The <code>beasvc</code> service was generating thread dumps when it received the <code>SERVICE_CONTROL_INTERROGATE</code> control code. This could cause unintended thread dumps when <code>beasvc</code> was interrogated by other tools.</p> <p>A code fix ensures that thread dumps are generated only when a custom control code is received.</p>	8.1 SP3
CR175607	<p>Installing WebLogic Server as a Windows service immediately after uninstalling it sometimes created wrong registry keys, which could lead to startup problems.</p> <p>A code fix ensure that the registry keys are flushed and properly closed.</p>	8.1 SP3

## Deployment

Change Request Number	Description	Release Fixed
CR127393	<p>A number of memory leaks were identified with this problem:</p> <ul style="list-style-type: none"> <li>• ClientRuntimeDescriptors of EO objects remained on the heap after undeploying application.</li> <li>• EJBCache objects were not cleaned up after undeploying of the application.</li> <li>• Runtime mbeans specific to WebAppServletContexts were not unregistered after undeploying an application.</li> <li>• KeepAliveCache's timer object held an application-specific classloader.</li> <li>• Descriptor objects were not unregistered after undeploying an application.</li> <li>• Application-specific classes were held by the Introspector caches after undeploying an application that uses struts.</li> </ul> <p>Code has been added to WebLogic Server that fixes these problems.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR128537	<p>The <code>weblogic.deployer</code> utility exited after 60 minutes with the following exception, even if the <code>Timeout</code> parameter was set to a higher value:</p> <pre data-bbox="292 487 1123 591">&lt;Error&gt; &lt;Deployer&gt; &lt;lcaix17.bea.com&gt; &lt;myserver&gt; &lt;ExecuteThread: '0' for queue: 'weblogic.kernel.System'&gt; &lt;&lt;WLS Kernel&gt;&gt; &lt;&gt; &lt;BEA-149014&gt; &lt;Target myserver is not defined - error null&gt;</pre> <p>The code was fixed to use the default timeout only if it is not provided in the <code>Deployer</code> options. This way deployment will not fail at the default 60 minutes, when the timeout is provided. Instead, it will wait for the timeout to expire.</p>	8.1 SP3
CR130062	<p>When the <code>weblogic.Deployer</code> command was used from a wrapper class, deployment to multiple domains could have produced deployment failures. This was due to RMI calls that were made without a properly authenticated subject on the thread.</p> <p>A code fix was made to keep the subject properly authenticated until the deployment activity is complete.</p>	8.1 SP3
CR130592 CR133154 CR135864	<p>For Managed Servers in Managed Server Independence (MSI) mode, the logic to determine the type of J2EE-module did not use the local or staged copy of the application. This problem surfaced in the MSI mode as a result of the server's inability to determine the J2EE-module type.</p> <p><code>stagingLocation</code> is now used to determine the type of J2EE-module if the original path does not exist in the <code>config.xml</code> file of the Managed Server</p>	8.1 SP3
CR132685	<p>An <code>EJBModule</code> maintains a <code>Map</code> that maps the implementation class with the list of all EJB names whose implementation classes are derived from the keyed implementation class. This map is used during redeployment.</p> <p>There were two bugs in this implementation where:</p> <ul data-bbox="292 1251 921 1312" style="list-style-type: none"> <li>• There was an unnecessary map maintained in <code>EJBModules</code>.</li> <li>• The same EJB name was added repeatedly to the list in the map.</li> </ul> <p>These issues lead the EJB container to consume more memory and caused the server to run out of memory.</p> <p>The unnecessary map was removed and the EJB container only adds EJB names to the list which are not already added.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR133764	<p>An application was successfully deployed as an exploded ear and the META-INF directory contains a <code>weblogic-application.xml</code> file where a listener-class is defined and a listener class was located in APP-INF directory. After an extension template was created using <code>config_builder.cmd</code>, the META-INF and APP-INF directories were missing from the created <code>.jar</code> file and the application could not be deployed.</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3
CR133864	<p>When <code>wlconfig</code> and <code>wldeploy</code> ant commands were used to deploy applications, the deployment tasks were hanging.</p> <p>A code fix ensures that the RMI queue threads no longer poll for task completion while deployment is running.</p>	8.1 SP3
CR134856	<p>The administration console did not correctly display the current status of a Web application deployed on a cluster after clicking Stop on the Deploy tab.</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3
CR134938 CR137390	<p>There was a classloader issue when undeploying applications. A J2EE container maintains an application-specific classloader and classloader tree, which are used for loading application classes and module-specific classes. When the undeploy command was issued on the application, it closed the classloader and classloader tree before rolling back the modules in it, and therefore got a <code>NoClassDefFoundError</code> errors for the classes that were referred from the <code>destroy()</code> method of the servlet for the first time, because <code>servlet destroy()</code> is called from the <code>rollback()</code> of that specific module.</p> <p>A code fix ensures that the container will not close the classloader tree before rolling back the modules in it.</p>	8.1 SP3
CR135145	<p>During deployment, an application that contains a Web application targeted to virtual hosts tries to search for an existing application on the default Web server even though it is not targeted to the default Web server. Hence it fails to deploy the component on the virtual host if there is another application deployed on the default Web server with the same context-path.</p> <p>The application no longer searches for the existing application on the default Web server when it is not targeted to default Web server.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR136885	<p>During server startup, all applications were staged. This included applications that were not needed by the current server. This was observed from the entries in the StagedTargets property in the Application stanza of the <code>config.xml</code> file and the stage directory had all available applications staged.</p> <p>A code change was used to resolve this issue. Only the necessary applications needed on the current server are staged.</p>	8.1 SP3
CR175858	<p>Staged directories were not removed when an application was removed using the Administration Console.</p> <p>A code fix ensures that the staged directories are removed after an application is removed.</p>	8.1 SP3

## EJB

Change Request Number	Description	Release Fixed
CR090405	WebLogic Server newly supports <code>java.net.URL</code> resource connection factory, so that an	7.0 SP6
CR091204	EJB can obtain a obtain an HTTP connection to a Web server external to the WebLogic Server environment using the <code>java.net.URL</code> resource connection factory.  WebLogic Server will support this in the following ways:  If the Bean provider provides a direct URL for the <code>res-type java.net.URL</code> , WebLogic Server will create a URL object with the <code>jndi-name</code> provided and bind the object to the <code>java:comp/env</code> . The bean provider declares the <code>jndi-name</code> as follows in that case in the <code>weblogic-ejb-jar.xml</code> :  <pre>&lt;resource-description&gt;   &lt;res-ref-name&gt;url/MyURL&lt;/res-ref-name&gt;   &lt;jndi-name&gt;http://www.rediff.com/&lt;;/jndi-name&gt; &lt;/resource-description&gt;</pre> If the bean provider provides a string that is not a valid URL, Weblogic Server treats it as an object that maps to a URL and is already bound in the JNDI tree. In that case, WebLogic Server binds a <code>LinkRef</code> with that <code>jndi-name</code> .  <pre>&lt;resource-description&gt;   &lt;res-ref-name&gt;url/MyURL1&lt;/res-ref-name&gt;   &lt;jndi-name&gt;firstName&lt;/jndi-name&gt; &lt;/resource-description&gt;</pre> The URL can then be accessed as follows inside the Bean code:  <pre>URL url = (URL) context.lookup("java:comp/env/url/MyURL"); connection = (URLConnection)url.openConnection();</pre>	8.1 SP3
CR103038	When the <code>&lt;allow-remove-during-transaction&gt;</code> element in the <code>weblogic-ejb-jar.xml</code> was set to "False", and an application attempted to remove a stateful session bean involved in a transaction, an inaccurate exception was thrown.  <code>weblogic.ejb20.locks.LockTimedOutException</code> is not the exception called for in the EJB 2.0 specification, and it was replaced with the exception required: <code>javax.ejb.RemoveException</code> .	8.1 SP3
CR108169	When an EJB deployment descriptor was edited using the Administration Console, an incorrect method name was introduced.  The correct method name has now been implemented.	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR109267	<p>When using entity EJBs with container-managed persistence and automatic key generation, the following error was thrown when using some XA JDBC drivers:</p> <pre>XAER_RMERR : A resource manager error has occurred in the transaction branch</pre> <p>A change to the automatic key generation code resolved the problem.</p>	8.1SP3
CR110440	<p>For an entity cache timeout with a combined (application level) cache, if the <code>idle-timeout-seconds</code> for one of the beans is set to 0, other beans with <code>idle-timeout-seconds &gt; 0</code> are also not removed from the cache after timeout.</p> <p>A code change involving the registration of <code>idle-timeout-seconds</code> resolved the problem.</p>	7.0 SP5 8.1 SP3
CR122524	<p>An incorrect string was passed to retrieve the <code>tableName</code> from the <code>DatabaseMetaData</code> if the <code>schemaName</code> was not specified with the <code>tableName</code>.</p> <p>Code was added to pass the correct string to retrieve the <code>tableName</code> from the <code>DatabaseMetaData</code> when the <code>schemaName</code> is not specified with the <code>tableName</code>.</p>	8.1 SP3
CR123213	<p>Stateless session beans were not returned to the free pool if they were invoked from the BMP <code>ejbRemove</code>. As a result, the pool reached maximum size and timed out while waiting to get an instance.</p> <p>A code change resolved the problem.</p>	8.1 SP3
CR125501	<p>WebLogic Server was throwing an <code>Illegal Reentrant call error</code> in some situations when a client removed an EJB and then created a new EJB with the same primary key.</p> <p>A code fix was implemented resolve this issue.</p>	8.1 SP3
CR127187	<p>A <code>CacheFullException</code> was thrown for <code>cmp11</code> beans because of back door database deletes.</p> <p>For EJB creates, WebLogic Server now checks if the bean exists in the cache before inserting it into the cache. This eliminates the possibility that the cache state will get out of sync with the database due to back door database updates.</p> <p>The cache is now remains consistent even after back door database updates.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR127397 CR131629	<p>This problem was there for Remote Objects using <code>ActivatableServerRef</code>. Entity beans use <code>ActivatableServerRef</code>.</p> <p><code>ActivatableServerRef</code> called <code>activate</code> to get the bean instance. after invoking the bean method it called <code>deactivate</code>. In <code>deactivate</code> we were releasing the bean to the pool. In some cases only <code>activate</code> was being called. Because <code>deactivate</code> was not being called <code>WebLogic Server</code> was not releasing the bean.</p> <p><code>WebLogic Server</code> no longer relies on <code>activate</code> and <code>deactivate</code> calls. The <code>ActivatableServerRef</code> now explicitly calls <code>notifyRemoteCallBegin</code> and <code>notifyRemoteCallEnd</code>.</p>	8.1 SP3
CR127872 CR135787	<p>A <code>java.sql.BatchUpdateException: ORA-01006</code> or <code>java.lang.ArrayIndexOutOfBoundsException</code> is thrown when <code>enable-batch-operations</code> is set to <code>true</code> for an EJB that uses <code>Optimistic Concurrency</code> and a batch of queries arrive such that a parameter is null in the predicated clause of one query and not null in the predicated clause of another query.</p> <p>A code fix provides a check to verify whether the queries are fit for batching. If they are not, batching is turned off and the queries are executed using multiple update statements. The following warning message is sent to the server log:</p> <pre data-bbox="229 973 1067 1168">&lt;Apr 27, 2004 11:25:12 AM PDT&gt; &lt;Warning&gt; &lt;EJB&gt; &lt;BEA-011078&gt; &lt;The CMP EJB UserSession has enable-batch-operations set to true, however the update queries are not suitable for batching. Batched operations are turned off for this batch and the queries shall be executed with mutliple update statements.&gt;</pre>	8.1 SP3
CR128063	<p>When an MDB suspended a transaction, if the transaction was later resumed, it was committed or rolled back using a different thread. This caused problems for some XA implementations.</p> <p>The MDB container now ensures that the transaction is started and committed using the same thread.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR128956 CR133894 CR154779	<p>When the domain had two JMS targets with the same JNDI name, MDB deployment was trying to create two MDB pools with the same name. As a result of this, it was trying to create two <code>MessageDrivenEJBRuntimeMBeanImpl</code> objects with same name, which was causing an <code>Instance Already Exists</code> exception.</p> <p>This was resolved by a code fix so that when the <code>&lt;provider-url&gt;</code> stanza is mentioned in the MDB descriptor, it is treated as any foreign JMS provider. Also, when performing the Distributed and/or Migratable destination optimizations, consideration is given only to the JMS target that is related to the current server or current cluster. No consideration is given to the JMS targets of other clusters.</p> <p>This way, MDB deployment will not fail when the JNDI name is reused for different JMS destinations across the domain in different clusters. Also, the <code>&lt;provider-url&gt;</code> stanza should not be mentioned in the MDB descriptors to have the Distributed and/or Migratable destination optimizations, as documented in the “<a href="#">MDB Deployment Options</a>” section of “<a href="#">Programming WebLogic Enterprise Java Beans</a>”.</p>	8.1 SP3
CR128980 CR135722	<p>When an MDB used the synchronous message polling scheme, it had issues when a TIBCO JMS server was used. Specifically, the MDB container’s polling optimizations could result as a delay in messages being received.</p> <p>The code was fixed so that the optimized poller is not used. Since the TIBCO message delivery scheme does not work well with this scheme, MDBs now use a poller that continuously polls the TIBCO JMS server. And since these pollers need an exclusive thread, when a <code>dispatch-policy</code> is specified with an MDB, the fix ensures there are enough threads for polling. Otherwise, the user can also tune the <code>max beans</code> count in the MDB descriptors.</p> <p><b>Note:</b> This fix only applies to TIBCO JMS, so other foreign JMS servers still use the existing mechanism. Therefore, there will not be any change in the behavior for MDBs using foreign JMS servers other than TIBCO.</p>	8.1 SP3
CR129035 CR132804	<p>For a <code>CMP20</code> bean using automatic key generation with the MS SQL Server database, and the <code>delay-db-insert-until</code> element set to <code>ejbPostCreate</code>, an entity local object that did not have the correct identity set got cached in the relationship set. This unidentified, entity local object, was returned when a getter method was called for the <code>CMR</code> field, thereby causing a <code>ClassCast</code> exception.</p> <p>This was resolved by associating a <code>dummyPK</code> to the entity object/entity local object before calling a <code>create</code> on the bean and replacing it with the correct identity when the <code>PK</code> is known.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR129185	<p>WebLogic Server loaded optimistic-concurrency beans in a separate transaction, suspending the current transaction, starting a new one, loading the bean and resuming the old transaction for all databases except Oracle, to avoid acquiring locks during the SELECT. Because the default behavior in Sybase is for a shared-lock to be acquired during the SELECT but released even before the statement has completed the statement, the loading behavior has been changed: When the concurrency-strategy is optimistic, WebLogic Server now suspends and resumes the transaction for databases other than Oracle only if the isolation-level is higher than Read-Uncommitted or Read-Committed.</p>	7.0 SP5 8.1 SP3
CR129223	<p>CMP20 EJBs using Optimistic concurrency and <code>cache-between-transaction</code> failed for many-to-many relationships.</p> <p>With the <code>cache-between-transactions</code> element set to true, the bean along with the generated many-to-many set was cached, which caused the new transaction to use the cached bean and hence the cached set. The previously cached values in the set were not cleared and hence at the end of the transaction when the M2NJoinTable inserts are done, it failed with a <code>ORA-00001: unique constraint violated error</code>. This indicated that an attempt was made to insert a record with a duplicate (unique) key.</p> <p>The problem was resolved by clearing the cached many-to-many set for the new transaction since the many-to-many set is maintained per transaction.</p>	8.1 SP3
CR129379 CR136730	<p>When an EJB transaction created many new entities or otherwise engaged many beans that all use JDBC, WebLogic Server risked running out of Oracle cursors, because in an attempt to avoid a suspected Oracle driver bug, WebLogic Server delayed closing JDBC statements until the end of a transaction, holding the cursors for the statement until then.</p> <p>This behavior has been changed so that the session need not hold cursors until the transaction ends.</p>	7.0 SP5 8.1 SP3
CR131848	<p>A <code>ClassCastException</code> was thrown at runtime when “<code>cache-between-transactions</code>” is set to true for a <code>BMP bean.java.rmi.RemoteException: EJB</code>.</p> <p>A check for BMP and CMP20 beans was added to prevent the <code>ClassCastException</code>.</p>	7.0 SP5 8.1 SP3
CR133602	<p>WebLogic Server used a special handling code for SonicMQ 4.* version, when XAconnections were used. This code was not required for Sonic 5.* version and caused a connection leak when used with it.</p> <p>A code fix eliminated the connection leak.</p>	7.0 SP5, 8.1 SP3

Change Request Number	Description	Release Fixed
CR133774	<p>The following part of the description for the <code>max-beans-in-cache</code> element in the <code>weblogic-application.xml</code> DTD was incorrect: If 0 is specified, then there is no limit for <code>max-beans-in-cache</code>. A <code>max-beans-in-cache</code> size of 0 would set the actual size of the cache to 0, thus causing a <code>CacheFullException</code>.</p> <p>The above note has been removed from the DTD. Also, a compliance check has been added to prohibit a value of 0 for the <code>max-beans-in-cache</code> in <code>weblogic-application.xml</code>. If you need to set a large size (infinite) for your cache, set the value of the <code>max-beans-in-cache</code> to <code>java.lang.Long.MAX_VALUE</code> in the DDs. This makes it compliant with the behavior of the same element in <code>weblogic-ejb-jar.xml</code>.</p>	8.1 SP3
CR134082	<p>The <code>trans-timeout-seconds</code> property was missing from the Marathon General Panel for session beans (both stateful and stateless) and for message-driven beans.</p> <p>The property has been added the General Panel for all EJBs and removed from the Persistence Panel for entity EJBs.</p>	8.1 SP3
CR135410	<p>When the bean is looked up and a business method is called for the first time, the call to <code>EntityContext.getCallerPrincipal()</code> from <code>ejbStore</code> returns the user as "anonymous". After this, when <code>ejbStore</code> is invoked again, it returns the correct username. This problem only occurs with <code>ejbStore</code> and no other method.</p> <p>The problem was resolved with a code change.</p>	8.1 SP2 8.1 SP3
CR174119	<p>When the <code>cache-between-transaction</code> element was set to true and a transaction was rolled back, the bean was marked as invalid but was still kept in the cache. Also, when <code>cache-between-transaction</code> was true, the <code>findByPrimaryKey</code> implementation checked whether the bean was already cached. Therefore, a bean could be reused from the cache even if it had been in an invalid state. It could also be returned and lazily loaded when a remote method was called. When the bean was loaded from the database, a <code>NoSuchObjectException</code> was thrown because the row did not exist in the database since the earlier transaction was rolled back. Therefore, an invalid bean was being made valid and returned from the cache.</p> <p>Now when <code>findByPrimaryKey</code> checks whether the bean is already in the cache, it checks only for ready beans and valid beans, and the cache check no longer returns an invalid bean. After not finding a bean in the cache, the <code>findByPrimaryKey</code> runs the query against the database. If it does not find the bean in the database, it throws a <code>FinderException</code>.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR174845	<p>Under some conditions, ReadOnly entity beans were not refreshed after the read-timeout-seconds value expired.</p> <p>A code change was made to initialize the bean, load from the <code>resultSet</code> and set the <code>lastLoadTime</code> only if the bean has timed out or has been invalidated.</p>	8.1 SP3

## Installation

Change Request Number	Description	Release Fixed
CR103125 CR172837	<p>The <code>"-log:C:/temp/xxx.log"</code> option was added to <code>beasvc</code>. It redirected all stdout/stderr to a log file. In production, this file grew quickly and became very large. On Windows, there was no way to copy or rename the file.</p> <p>Code was added to allow rotation of the log file on a Windows service.</p> <p>For additional information, see <a href="http://e-docs.bea.com/wls/docs81/adminguide/winservice.html">http://e-docs.bea.com/wls/docs81/adminguide/winservice.html</a></p>	6.1 SP6 8.1 SP3
CR107512	<p>There were issues with using Smart Update with a proxy. During the Smart Update process, the hard-coded proxy settings were not carried over or recognized.</p> <p>Proxy support is now enabled by automatically passing the following command-line system properties from <code>appBoot</code> to the Smart Update utility:</p> <ul style="list-style-type: none"> <li>• <code>http.proxyHost</code></li> <li>• <code>http.proxyPort</code></li> <li>• <code>http.proxySet</code></li> <li>• <code>https.proxyHost</code></li> <li>• <code>https.proxyPort</code></li> <li>• <code>https.proxySet</code></li> </ul> <p>Log priority settings are now also passed to Smart Update.</p>	8.1 SP3
CR121882	<p>The generic UNIX installer threw a misleading error message when the <code>tmp</code> filesystem was either too small or too full.</p> <p>The code was changed so that the <code>-Djava.io.tmpdir</code> property can be set to a filesystem with enough space.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR125514	<p>When the <code>config_builder.sh</code> script was run the Configuration Template Builder did not start up, and generated the following message: “Unable to instantiate GUI, defaulting to console mode”. However, it did not start up in Console mode and returned to the command prompt.</p> <p>This is the correct behavior. However, the Console now provides the following message: “The Configuration Template Builder is not supported in console mode.”</p> <p>The Configuration Template Builder can be invoked only in graphical mode. Therefore, the console attached to the machine on which you are running the Configuration Template Builder must support a Java-based GUI. All Windows-based consoles support Java-based GUIs; only a subset of UNIX-based consoles support Java-based GUIs. For more information, see “<a href="#">Starting the WebLogic Configuration Template Builder</a>”, in <i>Creating WebLogic Configurations Using the Configuration Wizard</i>.</p>	8.1 SP3

## J2EE

Change Request Number	Description	Release Fixed
CR107166	<p>Using a custom classloader structure with EJB implementation inheritance caused a <code>NoClassDefFound</code> error.</p> <p>This was resolved by setting the module classloader as the parent to the impl classloader.</p>	8.1 SP3
CR112618	<p><code>wlcompile</code> failed when the <code>weblogic.jar</code> was not in the <code>\$CLASSPATH</code>.</p> <p>Code was added to enable <code>wlcompile</code> to locate a <code>weblogic.jar</code> file referenced in an ant task's <code>taskdef</code>.</p>	8.1 SP3
CR125191	<p>Deployment could not properly resolve the application path if the path was relative and the JVM <code>user.dir</code> was not equal to the WebLogic root directory.</p> <p>Deployment now uses the <code>weblogic.RootDirectory</code> path to resolve relative paths.</p>	8.1 SP3
CR125191	<p><code>Bootstrap.apply()</code> was not used in <code>J2EEutils.getDeploymentCategory()</code>.</p> <p>If a path does not exist, <code>SlaveDeployer.resolveStagingPath()</code> is now used to resolve the path correctly, by invoking <code>Bootstrap.apply()</code>.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR126403 CR134398	<p>When an application with one hundred or more modules was deployed, the deployment activity was taking too much time. This increased the startup time for the server.</p> <p>The J2EEApplicationContainer code asserted for certain conditions and tried to frame a string as part of this activity.</p> <p>Code was added to avoid the formatting of unnecessary large strings during deployment.</p>	8.1 SP3
CR137465	<p>Because only a single class finder was used for all listeners, it was reset for every invocation. However, only the last listener's URI was available with the class finder.</p> <p>The code was changed so that a new class finder was created for every URI entry and is available for the application classloader.</p>	8.1 SP3
CR173871	<p>When compiling a large EAR file, the <code>appc</code> compiler ran out of memory.</p> <p>The compilation of the generated EJB files is now done inline by default. Before this change, the default behavior was to fork the compiler process (<code>javac</code> by default). This sometimes required <code>-J</code> options to be set in the <code>ExtraEJBOptions</code> so that they would be passed to the forked compiler process. With the default behavior changed to inline compilation, you no longer need to set the <code>-J</code> options in <code>ExtraEjbcOptions</code>.</p>	8.1 SP3

## JCOM

Change Request Number	Description	Release Fixed
CR095485	<p>A COM object that closed during a load could result in sockets being left in a <code>CLOSE_WAIT</code> state.</p> <p>The sockets now close on <code>endOfStream</code>.</p>	7.0 SP5 8.1 SP3
CR172074	<p><code>jCom</code> would throw a <code>NullPointerException</code> when the thread context object associated with a subject is null.</p> <p>This problem was resolved with a simple code fix.</p>	8.1 SP3
CR182512	<p>COM to Java clients failed when the Microsoft Security Bulletin MS04-011 was applied.</p> <p>This has now been resolved.</p>	8.1 SP3

# JDBC

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR099912	<p>HighestNumUnavailable defaulted to 0, which caused all idle connections to be locked for testing and released all at once. This condition held connections longer than necessary, and caused unexpected ResourceExceptions from pools that seemed to be sized correctly.</p> <p>New code makes the HighestNumUnavailable test and release connections one at a time. There is also a new configurable attribute added, secondsToTrustIdleConnection, which allows a connection to skip testing if it has been used within the set number of seconds.</p>	8.1 SP3
CR108826	<p>WebLogic Server Multipool did not provide an option to define time-based failover. The exact condition under which failover would be triggered to a secondary connection pool was uncertain. It would go through all the connections in the pool before it decided that it did not have a good connection. If the primary database went down, it needed to wait for xx time before the connection failed over to a secondary database.</p> <p>Code was added to provide an option to run a time check before returning the connection to the primary database. See <a href="#">“JDBC MultiPool Failover Enhancements” on page 1-21</a>.</p>	7.0 SP5 8.1 SP3
CR111231 CR128673	<p>In previous releases of WebLogic Server, memory leaks occurred when a BLOB column was used through a JDBC Tx Data Source.</p> <p>A code fix resolved the problem.</p>	6.1 SP6 7.0 SP4 8.1 SP3
CR120411	<p>When an applet made a connection via a datasource and the applet was killed, WebLogic Server printed leak detection messages. These messages were sometimes sent in error, and were not properly formatted for the JDBC/WLS log.</p> <p>JDBCLogger is now used to format the message in case of connection leak detection.</p>	8.1 SP3
CR120455	<p>A memory leak was discovered in WebLogic Server 6.1 Service Pack 2. The leak occurred when using a TxDataSource to access a BLOB column on a database with the WebLogic XA driver. This problem was solved with a code fix.</p>	6.1 SP6 8.1 SP3
CR120533	<p>When a JDBC log name was specified with the ".log" suffix, JDBC logging did not rotate as expected. The prior log file was overwritten by a new JDBC log when WebLogic Server was restarted.</p> <p>JDBCService.initLog() was modified to fix the problem.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR124017	<p>When PoolShrinking was enabled with an Informix database, XAConnectionPool failed with a SQLException.</p> <p>A JDBC XAConnection wrapper code change resolved the problem.</p>	8.1 SP3
CR124933	<p>An Oracle BLOB sometimes used a pooled connection after the connection pool determined that the connection was available for reassignment.</p> <p>Code was added to ensure the BLOB is completely processed before closing the pool connection or ending the transaction.</p>	8.1 SP3
CR125844	<p>Configuring idle-connection-timeout limits could result in unintended thread blocks or deadlocks because the JDBC pool may try to reclaim a connection that has not changed for the idle-timeout period but is still waiting for the DBMS to respond.</p> <p>A code fix was implemented to ensure that the JDBC pool correctly recognizes idle connections.</p> <p><b>Note:</b> Configuring idle-connection-timeout limits are not intended to aid in stopping runaway or deadlocked transactions.</p>	8.1 SP3
CR126808	<p>When two applications with one application-scoped datasource per application were deployed on the same server and the datasources have the same name, an InstanceAlreadyExistsException was thrown.</p> <p>Code was added to configure the pool name to be application-specific when necessary.</p>	7.0 SP5 8.1 SP3
CR126950 CR174756	<p>In an instance where DB side cursors were discarded, such as by running an Oracle STATS package for the cost based optimizer, the cached prepared statements for the JDBC connections on WLS side became invalid.</p> <p>There was no transparent way to have the prepared statement cache cleared by WebLogic Server in case exceptions were thrown due to this scenario. The cached statements remained invalid in the WebLogic Server cache until either WebLogic Server was restarted or the cache was cleared via the WebLogic Server Administration Console, or a JMX call.</p> <p>WebLogic Server now clears a prepared cache when an error occurs.</p>	8.1 SP3
CR127615 CR161930	<p>The connection pool held a "ConnectionEnv" object that did not refer to any connection directly, but contained a map with statements. Those statements had a reference to the connection that created them, and so were still holding on a hierarchy of database objects.</p> <p>Code was added to release the ConnectionEnv object in a timely manner.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR127720	<p>New versions of JDBC drivers track the transactional state of connections. If a local transaction was active on a connection, XA operations could not be performed on it, resulting in an XAER_PROTO or XAER_RMERR when an <code>xa_start()</code> was called on the connection. As a result, applications had to go through the tedious process of narrowing down where in their code they had started but not ended a local transaction.</p> <p>The problem was resolved by a code change in the recovery method that prevents special XA connections from being released to the pool twice.</p>	7.0 SP5 8.1 SP3
CR127891	The format of the connection leak file was modified to make the information more readable.	6.1 SP6 8.1 SP3
CR128117	<p>You can now configure a data source so that it binds to the JNDI tree with multiple names. You can use a multi-named data source in place of configuring multiple data sources that point to a single JDBC connection pool.</p> <p>For more details, see <a href="#">“Binding a Data Source to the JNDI Tree with Multiple Names”</a> in the Administration Console Online Help.</p>	8.1 SP3
CR128156 CR134273 CR161930	<p>The code that replaces a pooled JDBC connection did not completely remove all references to replaced objects, so memory grew with each replaced connection. The problem was noted when using the <code>getVendorConnection()</code> method.</p> <p>Now when WebLogic Server replaces a pooled connection, the connection is not referenced by other objects. This makes the replaced connection available for garbage collection.</p>	8.1 SP3
CR128419 CR132944	<p>During a graceful shutdown, the <code>BufferedOutputStream</code> to the <code>jdbc.log</code> was not completely flushed/closed, which could result in data loss in the buffer.</p> <p>The code was modified so that <code>BinaryOutputStream</code> is now <code>closed()</code> in the <code>JDBCService.prepareToSuspend()</code> method.</p>	8.1 SP3
CR128545	<p>Batched RowSet updates were failing on Informix if a table name included an upper-case letter. Informix uses lower-case letters for all metadata names. The WebLogic Server RowSet implementation relies on metadata for batch updates.</p> <p>The WebLogic Server RowSet implementation now converts metadata names to lower-case when using an Informix JDBC driver.</p>	8.1 SP3
CR130306	<p>In <code>jta.DataSource</code>, when doing <code>refreshAndEnlist</code>, WebLogic Server called <code>tx.enlist()</code>, but the connection was not returned to the pool if there was an exception in the <code>refreshAndEnlist</code> call.</p> <p>A code change catches the exception and releases the connection to the pool.</p>	7.0 SP5 8.1 SP3

Resolved Problems for Service Pack 3

Change Request Number	Description	Release Fixed
CR131575	<p>WebLogic Server sometimes threw a bogus warning about a JDBC connection leak:</p> <pre data-bbox="229 461 1056 534">[SerialConnection] : Connection Leak detected!!!!!!java.lang.Throwable: StackTrace at creation of connection: /n</pre> <p>The leak detection code that sent this warning is obsolete.</p> <p>A code change resolved the problem.</p>	7.0 SP5 8.1 SP3
CR131702	<p>During an attempt to reset a connection pool during high load, a deadlock was occurring between <code>DataSource.commit()</code> and <code>RESET_POOL</code>.</p> <p>In the <code>weblogic.jdbc.common.internal.ConnectionPool.reset()</code> method, the connection pool is now suspended before being refreshed, and is then resumed to avoid a deadlock.</p>	8.1 SP3
CR132293 CR134921	<p>For certain JDBC connections, the call to roll back a transaction was not being handled immediately because the driver had to wait for any currently-executing statement to return.</p> <p>In order to remove this delay, code was implemented to send a <code>Statement.cancel()</code> to abort any such executing statement when WebLogic Server is in the process of rolling back an ongoing transaction.</p>	8.1 SP3
CR132675	<p>The <code>isClosed()</code> method for all non-XA connections within a given transaction returned true after a <code>Connection.close()</code> call to any one of them.</p> <p>A code fix supplies a separate wrapper for the same underlying connection to all users involved in the transaction.</p>	8.1 SP3
CR132839	<p>When the test-connections-on-reserve is enabled and the isolation level is <code>TransactionReadCommitted</code>, a <code>SQLException</code> occurs when the customer is using the Ingres driver. The EJB has the <code>TRANSACTION_READ_COMMITTED</code> transaction isolation level. When calling the EJB, the database connection fails on the following <code>SQLException</code>:  "SQLState(25000) vendor code(5932) ca.edbc.util.EdbcEx: SET TRANSACTION: This statement is not allowed.</p> <p>No change, except that those calls which would fail on a connection that already held DBMS locks should now succeed.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR134364	<p>A synchronization issue (not a deadlock) occurred when the ResourcePoolMaintenanceTask attempted to time out all inactive resources. It got a lock to the ConnectionPool and proceeded to check all the connections. However, if a connection was active at that point and locked on another object, the ResourcePoolMaintenanceTask had to wait to get that lock. This caused a slowdown and a "frozen" server to the end user. The task completed but took a very long time.</p> <p>A code change prevents the connection pool from locking the entire connection pool. Another method to avoid this is to turn off the reclaim process, by setting the timeout value to zero.</p>	8.1 SP3
CR135116 CR172353	<p>WebLogic Server unexpectedly changed autocommit mode for a JTS connection. This occurred when WebLogic Server failed connection testing on reserve. Because a JTS connection accepts global transactions and a connection is in the user transaction, it should honor the user transaction boundary. Furthermore, connection testing should be implicit behavior for users. This has not been the case in prior releases.</p> <p>The problem was resolved with a code change.</p>	8.1 SP3
CR136624	<p>When Connection Creation Retry Frequency on a JDBC connection pool was set to enable connection retries (not set to 0), the connection creation retry logic was invoked when an application attempted to reserve a connection. If the database was unavailable, the thread would hang during the wait and retry cycles.</p> <p>Connection reserve requests now succeed or fail quickly as in previous releases.</p>	8.1 SP3
CR136681	<p>The wiconfig ant task did not properly remove a pool. The problem is within the JDBC connection manager code.</p> <p>The problem was resolved with a code change.</p>	8.1 SP3
CR136746	<p>Enhancement request to allow explicit naming of JDBC driver being used in class VendorId. This customer enhancement was made through a code change.</p>	6.1 SP7 8.1 SP3
CR137180	<p>Changes in weblogic.jdbc.vendor.oracle.OracleStatement from 81SP1 to 81SP2 broke Oracle 9 Driver compatibility. The return type for the method getDBDescription() on Interface weblogic.jdbc.vendor.oracle.OracleStatement changed from DBColumn[] in 81SP1 to Object[] in 81SP2. This change made 81SP2 incompatible with Oracle 9 Drivers when a customer would like to wrap the java.sql.Statement object with a dynamic proxy.</p> <p>This problem was resolved through a code change. In OracleStatement.java, references to the following two unsupported methods were removed: getDBDescription and getBinds.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR174025	<p>In response to requests for a “Purge Policy” for JDBC connection pools, the <code>StatementTimeout</code> and <code>TestStatementTimeout</code> attributes were added to JDBC connection pools to limit the amount of time that a statement can execute on a database connection from a JDBC connection pool.</p> <p>For detailed information, see “<a href="#">JDBC Connection Pool Testing Enhancements</a>” in <i>Programming WebLogic JDBC</i>.</p>	8.1 SP3
CR174126	<p>Using multiple different prepared statements in a single transaction invoked a bug in the XA statement cache, which lost the handle to unclosed JDBC statements. This quickly caused all the available cursors in Oracle to be consumed and leaked.</p> <p>A fix was made to the underlying collection object being used to implement the XA statement cache.</p>	8.1 SP3
CR174575 CR175701	<p>A JDBC Multipool fails with an <code>IndexOutOfBoundsException</code> under a heavy load due to a non-threadsafe manner of incrementing an array index.</p> <p>The code was repaired to be threadsafe.</p>	8.1 SP3
CR174658	<p>When a Java client that used Weblogic datasources experienced a “PeerGone” exception, the transaction contexts related to the connection were not properly cleaned up.</p> <p>The problem was resolved with a code fix.</p>	8.1 SP3
CR177220	<p>In a multi-server domain, a client involved in a global transaction got the following exception:</p> <pre data-bbox="229 1098 1032 1150">weblogic.transaction.RollbackException: delist() failed on resource jdbcXAPool</pre> <p>The code was changed so that if the data source is null, then the resource manager is not set. When the data source is initialized, the resource manager is wrapped by <code>VendorXAResource</code>.</p>	8.1 SP3
CR180097	<p>The JDBC connection pool’s “CountOffTestFailuresTillFlush” property neglected to clear statement caches.</p> <p>The problem was resolved with a code fix.</p> <p>For detailed information about using the “flush pool” feature, see “<a href="#">JDBC Connection Pool Testing Enhancements</a>” in <i>Programming WebLogic JDBC</i>.</p>	8.1 SP3
CR208038	<p>When a JDBC connection pool was created using the silent script mode, with the <code>MaxCapacity</code> parameter set to 1, the Administration Console incorrectly displayed the parameter’s value as 15.</p> <p>This problem has been resolved.</p>	8.1 SP3

# jDriver

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR113462	WebLogic Server threw a NumberFormatException when using BigDecimal types in an environment with Oracle's NLS_LANG setting. The problem did not occur when using Oracle's NLS_NUMERIC_CHARACTERS parameter to match American-style numeric character definitions. This problem was solved with a code fix.	6.1 SP6 8.1 SP3
CR125135	By default, WebLogic jDriver for Oracle/XA Data Source set the value of the oracleXATrace parameter to true, rather than false. This caused the driver to create trace files of the form xa_poolname*.trc that could grow large over time, unless you specifically disabled trace files by setting oracleXATrace="false" in config.xml. The code was modified to set the default value of oracleXATrace to false if no value is specified.	7.0 SP5 8.1 SP3
CR125135	By default, WebLogic jDriver for Oracle/XA Data Source set the value of the oracleXATrace parameter to true, rather than false. This caused the driver to create trace files of the form xa_poolname*.trc that could grow large over time, unless you specifically disabled trace files by setting oracleXATrace="false" in config.xml. The code was modified to set the default value of oracleXATrace to false if no value is specified.	7.0 SP5 8.1 SP3
CR127949	Statement.getResultSet() sometimes generated an unnecessary new ResultSet wrapper. A code change has resolved the problem.	6.1 SP6 8.1 SP3
CR129220	WebLogic Server Oracle jDriver was not properly releasing CLOB objects for garbage collection. A code change resolved the problem.	6.1 SP6 8.1 SP3
CR129220	WebLogic Server Oracle jDriver was not properly releasing Clob Objects for garbage collection. A code change resolved the problem.	7.0 SP5 8.1 SP3

Change Request Number	Description	Release Fixed
CR134285 CR137123 CR181738	<p>The <code>weblogic.db.oci.OciLob</code> class defined two class level variables to hold the returned <code>byteArray</code> for either BLOB or CLOB data from the database. As this byte array could be quite large each instance of <code>OciLob</code> filled up the heap quickly until a garbage collection occurred. Further testing and a modified <code>OciLob</code> moving the class level variables <code>retBArray[]</code> and <code>retCArray[]</code> into the methods caused the variable to be allocated on the stack and reduced the size of the <code>OciLob</code> object instance and thus overall heap usage was reduced.</p> <p>Code was added to move the global variables <code>retBArray</code> and <code>retCArray</code> to the method level in order to reduce the size of the memory heap usage.</p>	6.1 SP? 7.0 SP5 8.1 SP3
CR136168	<p>WebLogic <code>jdbcDriver</code> may cause the server to crash with Oracle error message <code>ORA-02392</code> when using the long raw type under heavy loads.</p> <p>A code fix was implemented to resolve this issue.</p>	8.1 SP3
CR142730	<p>After a long database outage, a JDBC connection pool that used the <code>XAjdbcDriver</code> for Oracle with <code>TestConnectionsOnReserve="true"</code> could not recover and recreate connections to the database. The following error messages were thrown:</p> <pre data-bbox="229 942 1063 1078">&lt;Warning&gt; &lt;JDBC&gt; &lt;001096&gt; &lt;Refreshing this bad pool connection failed weblogic.common.ResourceException: java.sql.SQLException: open failed for XAResource 'oracleXAPool' with error XAER_RMERR : A resource manager error has occurred in the transaction branch.</pre> <p>and</p> <pre data-bbox="229 1133 1063 1237">&lt;Warning&gt; &lt;JDBC&gt; &lt;001096&gt; &lt;Refreshing this bad pool connection failed weblogic.common.ResourceException: java.sql.SQLException: LDA pool exhausted - make sure you call Connection.close()</pre> <p>The problem was that during the outage, the JDBC connection pool attempted to recreate connections, but on failure, those connection attempts were not cleaned up and were depleting Oracle client resources (in the LDA).</p> <p>The <code>jdbcDriver</code> now cleans up connection creation attempts that fail.</p>	8.1 SP3
CR172462	<p>The WebLogic Server <code>jdbcDriver</code> was not functioning properly with Oracle 9.2 when using the <code>AL32UTF8</code> character set.</p> <p>This problem was resolved with a code fix.</p>	7.0 SP5 8.1 SP3

# JMS

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR110911	Ordered redelivery did not always work correctly when a new message was added to the queue.  Code was added and ordered redelivery now works correctly when the restrictions are followed.	8.1 SP3
CR110991 CR117044	The <code>JMSServerRuntimeMBean</code> method <code>getHealthState</code> is no longer a public method.	7.0 SP5 8.1 SP3
CR112112 CR135792 CR181852	During server startup, JMS code created <code>DestinationRuntimeMBeans</code> and did not register them with the <code>MBeanServer</code> . Before these <code>destinationMBeans</code> were registered, it then created and registered <code>DurableSubscriberRuntimeMBeans</code> (children of <code>DestinationMBeans</code> ) for subscribers that were persisted in the <code>FileStore</code> . The subscribers then had no parent component in their names and fell out of the scope of the parent when the console scoped them.  WebLogic Server now waits and registers the destination's durable subscribers after the destination has been registered, but before the destination is advertised. Durable Subscribers are not dropped from the console when monitoring JMS Destinations.	8.1 SP3

Change Request Number	Description	Release Fixed
CR121041	<p>A problem occurred with a Messaging Bridge between two JMS topics running in two Managed Servers in the same domain, running on the same machine. As the Messaging Bridge was started during WebLogic Server startup, it threw the following warning message. Messages could not be forwarded.</p> <pre data-bbox="229 548 1063 968">####&lt;Aug 20, 2003 12:49:11 PM MST&gt; &lt;Warning&gt; &lt;MessagingBridge&gt; &lt;slsol4.bea.com&gt; &lt;server2&gt; &lt;ExecuteThread: '10' for queue: 'default'&gt; &lt;kernel identity&gt; &lt;&gt; &lt;200026&gt; &lt;Bridge "Test Messaging Bridge" encountered some problems to one of its adapters or underlying systems. It stopped transferring messages and will try to reconnect to the adapters shortly. (Exception caught was java.lang.Exception: javax.resource.spi.IllegalStateException: Managed connection is closed at weblogic.jms.adapter.JMSManagedConnection.checkIfDestroyed(JMSManagedConnection.java:263) at weblogic.jms.adapter.JMSManagedConnection.sendEvent(JMSManagedConnection.java:231) at weblogic.jms.adapter.JMSBaseConnection.throwResourceException(JMSBaseConnection.java:1266) at ...</pre> <p>Analysis revealed that the bridge failed to create the message listener because it was configured for durable topics and there was no JMS store available. The Bridge encountered an internal error when trying to log the resource exception so the customer was not able to tell why the bridge was failing.</p> <p>The problem was resolved by a code change to allow the bridge to throw the correct resource exception. Now, the correct exception is logged by the bridge.</p>	7.0 SP5 8.1 SP3
CR121226	<p>The Messaging Bridge failed with a NullPointerException on startup when migrated.</p> <p>The migratable code was modified to properly initialize before starting so that the Messaging Bridge can now successfully migrate without errors.</p>	8.1 SP3
CR121741	<p>Inconsistent behavior occurred when shutting down the MQ-WLS bridge using MBeans. The exception occurred because <code>Recover()</code> is not a valid operation on a transacted session.</p> <p>A code change verifies that a session is not transacted before calling <code>recover()</code>.</p>	7.0 SP5 8.1 SP3

Change Request Number	Description	Release Fixed
CR121760	<p>An <code>InvalidDestinationException</code> was received when using WebLogic Server Messaging Bridge to integrate WebMethods 6.0 JMS with WebLogic Server. When sending messages to WebLogic Server, WebMethods 6.0 requests an <code>ack</code>. The bridge sends <code>ack</code> of type <code>weblogic.jms.DestinationImpl</code> when WebMethods expected type <code>com.wm.broker.jms.Destination</code>. An exception resulted.</p> <p>As WebLogic Server is providing bridge functionality between the two JMS implementations, the request is to convert message acknowledges to the correct type. It seems that this is correctly done for regular messages.</p> <p>Analysis revealed that the standard WebLogic JMS client was handed a WebMethods message to send, and attempted to call <code>setJMSDestination</code> on the WebMethods message using a WebLogic destination. WebMethods throws an <code>InvalidDestinationException</code> when this occurs.</p> <p>The problem was resolved by a code change to catch and ignore the <code>InvalidDestinationException</code>.</p>	7.0 SP5 8.1 SP3
CR123194	<p>In previous WebLogic Server service packs, when the server instance went down but its clients remained active, JMS threw a runtime exception <code>weblogic.rmi.extensions.RemoteRuntimeException</code>, instead of a <code>JMSException</code> as expected per the JMS specifications.</p> <p>A code change resolved the problem.</p>	7.0 SP5 8.1 SP3
CR124286 CR128684	<p>When MDB lost the connection to MQSeries because MQSeries had been shut down, the resource was not unregistered and MQSeries was left with prepared transactions that were not resolved until WebLogic Server was restarted.</p> <p>The <code>JMSConnectionPoller</code> was changed to call <code>unregisterResource</code> if it was disconnecting from JMS due to a connection failure. If it was disconnecting for any other reason (undeployment, migration, shutdown) then <code>registeredResource</code> was set to null.</p> <p>Transaction recovery now occurs correctly if MQSeries is stopped and restarted while the MDB connected to MQSeries is processing messages without having to restart WLS.</p>	8.1 SP3
CR125014 CR171724	<p>When a JMS client was repeatedly connected and disconnected to a JMS server, a memory leak occurred on the server side that resulted in <code>OutOfMemoryError</code>.</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3
CR125693	<p>The legal minimum value of the <code>flowMinimum</code> attribute of the <code>JMSConnectionFactoryMBean</code> had been changed to 1 and so, to be consistent, the legal minimum value of the <code>flowMaximum</code> attribute was also changed to 1.</p>	7.0 SP5 8.1 SP3

Change Request Number	Description	Release Fixed
CR125979	<p>Customers who set up a JMS bridge within a cluster sometimes found that when starting the bridge, the following exception was thrown from the Managed Servers, and that the JMS message could not be sent to the intended destination.</p> <pre data-bbox="229 517 1063 939"> &lt;Oct 16, 2003 3:12:49 PM PDT&gt; &lt;Notice&gt; &lt;WebLogicServer&gt; &lt;BEA-000360&gt; &lt;Server started in RUNNING mode&gt; &lt;Oct 16, 2003 3:23:17 PM PDT&gt; &lt;Error&gt; &lt;JMS&gt; &lt;BEA-040368&gt; &lt;The following exception has occurred: java.lang.NullPointerException at weblogic.jms.bridge.internal.MessagingBridge.startInternal(M essagingBridge.java:519) at weblogic.jms.bridge.internal.MessagingBridge.execute(Messagi ngBridge.java:956) at weblogic.kernel.ExecuteThread.execute(ExecuteThread.java:197 ) at weblogic.kernel.Kernel.execute(Kernel.java:336) at weblogic.kernel.Kernel.execute(Kernel.java:321) at ... </pre> <p>A code fix was implemented to ensure the Bridge starts correctly in clustered configurations where a Managed Server's Started attribute is changed from <i>false</i> to <i>true</i> and no adapter has been deployed.</p>	8.1 SP3
CR126183	<p>An idle bridge was logging a message after the maximum idle time setting had been reached. Code was added to suppress the repetitive log message "Bridge X start transferring messages" logged by an idle bridge.</p> <p>If the bridge is stopped and restarted, or if it encounters an exception and is restarted you will see the "Bridge &lt;bridgename&gt; starts transferring messages" log message, but you will not see the repetitive message logged by an idle bridge.</p>	7.0 SP5 8.1 SP3
CR126671	<p>The Message Bridge did not register a notification listener when a bridge adapter was deployed to a cluster.</p> <p>This problem was corrected with a code fix.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR127260	<p><code>MessagingBridgeRuntime</code> Means was not stopped by the <code>weblogic.Admin</code> utility command:</p> <pre>java weblogic.Admin -url t3://127.0.0.1:7001 -username weblogic -password weblogic INVOKE -type MessagingBridgeRuntime -method stop</pre> <p>The <code>MessagingBridge</code> start and stop methods now throw an informative exception about how to start and stop <code>MessagingBridge</code> at runtime.</p>	7.0 SP5 8.1 SP3
CR128596	<p>Message flows sometimes stalled when messages were sent over the Messaging Bridge using distributed destinations from a 7.0 SP2 cluster to 8.1 SP2 cluster when one of the 8.1 SP2 cluster servers was restarted.</p> <p>A code change resolved the problem.</p>	7.0 SP5 8.1 SP3
CR132606	<p>Messages Pending counters in the Administration Console were incorrect for Distributed Topic members that were on the same WebLogic Server instance. Counters were correct for remote distributed topic members.</p> <p>Code was changed to correctly manage the distributed topic reference count when forwarding to other non-remote distributed topic members.</p>	8.1 SP3
CR133155	<p>WebLogic Server took too long to recover JMS messages from the JDBC store at boot time. Including JMS in the <code>getTables</code> prefix resolved the problem.</p> <p>A code change solved the problem.</p>	8.1SP3
CR133474	<p>Messages were not dispatched to MDBs properly after a migrated JMS server was re-migrated back to the original host server instance.</p> <p>A code change solved the problem.</p>	8.1 SP3
CR134155	<p>JMS messages from topic sessions caused a memory leak when the receiver was created using a transacted session, used <code>AUTO-ACKNOWLEDGE</code> mode, and the message was rolled back.</p> <p>A code fix resolved this issue.</p>	8.1 SP3
CR137145	<p>Some JMS thin clients using IIOP were hanging because the server used the same lock when performing registrations and when notifying callback due to heartbeat failures.</p> <p>This problem was resolved with a code change.</p>	8.1 SP3
CR172511	<p>Closing a <code>JMSConnection</code> session before closing all <code>JMSConsumer</code> clients first could cause a memory leak.</p> <p>A code change solved the problem.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR172780	<p>Restarting a Tibco Server prevented a wrapped JMS object (ForeignJMSConnectionFactory) from sending any further messages, and threw the following security exception:</p> <pre>weblogic.management.NoAccessRuntimeException: Access not allowed for subject: principals=[], on ResourceType: ForeignJMSConnectionFactoryConfig Action: read, Target: Password weblogic.management.NoAccessRuntimeException: Access not allowed for subject: principals=[], on ResourceType: ForeignJMS ConnectionFactoryConfig Action: read, Target: Password</pre> <p>The problem was resolved by a code fix to the JMSConnectionFactory API so that it is aware of the KERNEL_ID for access to the username and password needed by the ForeignJMSConnectionFactory when reconnecting.</p>	8.1 SP3
CR173565	<p>JMS messages were delivered twice when an XA transaction failed.</p> <p>A code fix was made to handle failed transactions, and to log an appropriate error message to notify the user that there is a transaction in an ambiguous state.</p> <p>Prior to this change, any message involved in a transaction that was in an ambiguous state would still be rolled back and redelivered because the message was in memory. However, the record in the store (if there was one) was not properly updated because the store's handle for that record was invalid, leaving the record (if there was one) in the store. Therefore, when the JMS server was rebooted the message was redelivered, resulting in duplicate message.</p> <p>Now any message involved in a transaction that is in an ambiguous state will require a JMS server restart before it can be recovered. Any attempts to recover the message without JMS server restart will result in an RMERR.</p>	8.1 SP3
CR176366	<p>When the getObject() method is set on the Boolean.TYPE in the onMessage() API, it throws the following JMSEException:</p> <pre>javax.ejb.EJBException: nested exception is:weblogic.jms.common.JMSEException: Error deserializing object weblogic.jms.common.JMSEException: Error deserializing object</pre> <p>The problem was resolved with a code fix to the weblogic.jms.common.resolveClass() API.</p>	8.1 SP3
CR178405	<p>During a scan for expired messages, a java.util.ConcurrentModificationException was thrown.</p> <p>The problem was resolved with a code fix.</p>	8.1 SP3

## JNDI

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR110136 CR176352	<p>The NamingException was thrown as part of NestedException from the server when the client could not correctly deserialize the contents of the NamingException. The NamingException contained the WLCContextImpl object and for this object to be deserialized, WebLogic Server needed the environment to be on the thread for load balancing.</p> <p>This problem was fixed by initializing a default environment if an environment is not found on the thread .</p>	8.1 SP3
CR125793 CR185265	<p>When trying to obtain an InitialContext, WebLogic Server spent a significant amount of time trying to obtain the stub name.</p> <p>Code was added to ensure that WebLogic Server generates the stub name statically from PeerInfo.</p>	8.1 SP3

## JSP

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR104429 CR134313	<p>If a JSP was updated and manually recompiled, and the updated JSP class file was copied to the <code>jspWorkingDir</code>, WebLogic Server recompiled updated JSP pages without checking to see if recompilation was necessary.</p> <p>A new parameter, <code>-Dweblogic.jsp.alwaysCheckDisk</code>, when set to true causes WebLogic Server to check the JSP class on disk to determine if the JSP must be recompiled. This parameter is set to false by default, so default behavior is not changed.</p>	7.0 SP5 8.1 SP3
CR107421	<p>When a JSP containing <code>jsp:useBean</code> used an inner class, the compilation failed.</p> <p>Always use the '\$' class separator for an inner class within <code>jsp:useBean</code>. The JSP container has been modified to internally generate 'types' which always contain the '.' separator instead of the '\$'.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR111423	<p>When packaged in a WAR file, precompiled JSPs stored in a sub-context of the application were always recompiled upon deployment to WebLogic Server. This problem did not occur when JSPs were deployed from an exploded archive directory, or for JSPs in the root-context of a WAR file.</p> <p>The problem occurred because of a difference in the rounding behavior of timestamps used in the jar and zip formats. The discrepancy in rounding could cause an older timestamp (by one second) to be recorded in class files inside the WAR file, triggering the server to recompile the classes.</p> <p>The code was modified to advance the timestamps in compiled JSP classes by one second, thereby preventing JSPs from being recompiled.</p>	6.1 SP6 8.1 SP3
CR111655	<p>Previously, it was not possible to use JavaServer Pages Standard Tag Library (<b>JSTL</b>) tags in JSPs that include Japanese characters. When such a JSP is executed, an error starting with the following lines occurred:</p> <pre data-bbox="229 857 1002 965">java.io.IOException: javax.servlet.jsp.JspException: The taglib validator rejected the page: "org.xml.sax.SAXParseException: An invalid XML character (Unicode: 0x82) was found in the CDATA section."</pre> <p>If the <code>pageEncoding</code> attribute was not specified in the page directive, the byte stream, which is used for taglib validation, was constructed using default encoding.</p> <p>The problem was corrected with a code change to ensure use of the character encoding defined in the <code>contentType</code> attribute if the <code>pageEncoding</code> is not specified.</p>	7.0 SP5 8.1 SP3
CR112794	<p>A JSP was refreshed with a copy that did not parse correctly, and this caused a deadlock.</p> <p>The problem was resolved by a code change that added a check during JSP refresh.</p>	7.0 SP5 8.1 SP3
CR120914	<p>When you used the WebLogic Server form validation tag library, request parameters were not available to subsequent JSPs. This problem was solved with a code fix.</p>	6.1 SP6 8.1 SP3
CR123659	<p>WebLogic Server did not allow comments in JSP tags.</p> <p>Code was added to parse java/xml comments in JSP tags, so that xml comments are ignored in the generated java file and java comments are kept.</p>	8.1 SP3
CR124505	<p>A JSP with pre-compiled classes in a WAR file was being recompiled when it should not have been recompiled. Time stamp differences of up to 1.75 seconds were noted.</p> <p>New time buffer logic was applied and the time buffer was changed from 1 to 2 seconds. JSPs in WAR files with precompiled classes are no longer recompiled.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR127515 CR068074	<p>When there was a javac error statement on a jsp file, the jsp container displayed the error message correctly on both the browser page and the log file.</p> <p>When weblogic.jspc was used, the message became corrupted.</p> <p>A code change corrected the problem.</p>	8.1 SP3
CR127944	<p>In jspc code WebLogic Server parsed the JSP descriptor options provided by the user. One such option was the compileFlags attribute that suggested what type of compiler to use.</p> <p>The flag is denoted by the string compileFlags in weblogic.xml. WebLogic Server checked for the string compilerFlags; therefore the descriptor element was ignored.</p> <p>Code was added to change the string comparison from compilerFlags to compileFlags in jspc code. When you specify a compiler flag with the compileFlags option, it will be honored by jspc.</p>	8.1 SP3
CR130057	<p>When the wlpkg Ant task was used to build an exploded EAR application, it was not preserving the time stamp of the JSP files. This caused needless recompilation of JSPs even when they had already been compiled.</p> <p>This issue was resolved with a code fix so that wlpkg preserves the timestamps of the files that it copies.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR130438	<p>A translation error occurred with taglib and CDATA in a JSP page, as shown in this snippet:</p> <pre>sample.jsp : &lt;%@ page contentType="text/xml" %&gt; &lt;?xml version="1.0" encoding="iso-8859-1" ?&gt; &lt;%@ taglib uri="http://java.sun.com/jstl/core"; prefix="c"%&gt;  &lt;ablock&gt; &lt;![CDATA[ This is a cdata block ]]&gt; &lt;/ablock&gt;</pre> <p>When the JSP is compiled using the <code>weblogic.jspc</code> compiler, here is the output:</p> <pre>"sample.jsp": Translation of /sample.jsp failed: javax.servlet.jsp.JspException: The taglib validator rejected the page: "org.xml.sax.SAXParseException: The element type "jsp:text" must be terminated by the matching end-tag "&lt;/jsp:text&gt;"., "</pre> <p>A code fix checks the existence of CDATA while converting the content in XML syntax to a JSP document.</p>	8.1 SP3
CR133172 CR178872 CR179626	<p>The <code>jspc</code> compiler's "compileAll" option was ignoring JSPs under the WEB-INF directory. This was resolved with a code fix.</p>	8.1 SP3
CR133453 CR177115	<p>Pre-compiled JSPs were failing the stale checks. This was resolved with a code fix that adds logic for stale checks.</p>	8.1 SP3
CR172250	<p>There was no backward compatible property for <code>weblogic.jspc</code>. A new command line flag was added for <code>weblogic.jspc</code> called <code>backwardcompatible</code>. If it is set to true, then the Web application will be compiled maintaining backward compatibility with WebLogic Server 6.x and earlier. With the new "backwardcompatibility" flag, you can compile Web applications as in WebLogic Server 6.x.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR172380	<p>A struts-based application failed to pre-compile an EAR file, and throwing the following exception:</p> <pre>&lt;Error&gt; &lt;Deployer&gt; &lt;149205&gt; &lt;The Slave Deployer failed to initialize the application dos due to error weblogic.management.ManagementException: 149233 - with nested exception: [java.lang.ExceptionInInitializerError]. java.lang.ExceptionInInitializerError: java.lang.NullPointerException</pre> <p>A code change resolved the problem by setting the thread class loader before precompiling all the JSPs to the <code>ServletClassLoader()</code> method. This ensures that the same classloader gets used for loading the struts classes and loading any other class that the struts library tries to load afterwards.</p>	8.1 SP3
CR175937	<p>In the exploded webapp format, JSPs were recompiled when moved from one time zone to another.</p> <p>Stale-checking logic was added for exploded webapps.</p>	8.1 SP3

## JTA

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR112772	<p>Transactions were timing out at 30 seconds with out regard to the configured value of the transaction timeout attribute in the <code>config.xml</code> file.</p> <p>A code fix was implemented to ensure that WebLogic JTA utilizes the configured transaction timeout value at startup.</p>	
CR113226	<p>When a resource name contained more than 64 characters, WebLogic Server could throw an exception when testing the connection pool.</p> <p>The problem occurred because only the first 64 characters were tested for uniqueness. The code was modified to properly handle resource names longer than 64 characters.</p>	7.0 SP5 8.1 SP3

Change Request Number	Description	Release Fixed
CR122212	<p>When a timeout exception occurred within a Usertransaction it was marked for rollback. The actual rollback was scheduled by a timer and happened asynchronously.</p> <p>If in this situation the customer called <code>rollback()</code> explicitly on the transaction, the call did not directly issue the rollback or wait for the timer to perform the rollback. Directly calling <code>begin()</code> again lead to an <code>IllegalStateException</code> as the thread was still associated with the transaction to be rolled back by the timer.</p> <p>The call to <code>rollback()</code> should only return when the rollback has been performed. In this scenario <code>rollback()</code> returned immediately and no manual rollback operation was started as the transaction was internally marked to be rolled back by the timer.</p> <p>Code was added to ensure that in cases where the <code>rollback()</code> method is called due to other exception conditions, the transaction is now immediately removed from the thread and <code>begin()</code> can again be called immediately.</p>	8.1 SP3
CR122842	<p>When registering a new resource under the same name as the old resource, any subsequent XA transaction operations could not go to the new resource. They continued to attempt to reach the original resource.</p> <p>Code was added to allow the substitution of a second resource with the same name.</p>	7.0 SP3 8.1 SP3
CR124503	<p>WebLogic Server was synchronized on the Transaction Manager rather than the transaction itself, so the method that should have been called was <code>setRollbackOnly</code> instead of <code>setRollbackOnlyUnsync</code>.</p> <p>Code was added to make <code>setRollbackOnly</code> synchronous.</p>	8.1 SP3
CR173964	<p>In XA transactions with WebLogic JMS and Oracle XA as resource managers, killing the connection to Oracle at the moment where <code>Xa.prepare</code> is being done can lead to <code>XAER_RMFAIL</code> being returned, which means that WebLogic Server leaves transactions in doubt on the Oracle side. In this situation, Oracle has a global transaction created in a prepared state, but it is never being rolled back by WebLogic Server, thus causing the table being locked and subsequent calls returning with <code>ORA-1591</code> errors.</p> <p>A code change ensures that if <code>RMERR</code> or <code>RMFAIL</code> are received at <code>Xa.prepare</code>, then rollback is allowed to be called on the resource.</p>	8.1 SP3
CR177865	<p>The JTA recovery process stops the server from shutting down gracefully.</p> <p>This was resolved by changing the shutdown order of the JTA, JMS, and JDBC services (in that order) during server shutdown.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR182243	<p>A JDBC connection leak occurred when a connection was created under a transaction and the enlist with the current transaction failed; that is, the connection was not returned to the pool.</p> <p>A code fix resolved the problem.</p>	8.1 SP3

## JVM

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR125688 CR177828	<p>When using the JRockit JVM, calling <code>java.io.File.getCanonicalPath()</code> when using a UNC path, the following error was thrown:</p> <pre>java.io.IOException: \\UNC path: The file name, directory name, or volume label syntax is incorrect.</pre> <p>This error was noted when using the <code>DeployerRuntimeMBean</code> to deploy an EAR file with an application source file path listed as a URL in the descriptor file, such as <code>file://localhost/D:/myfolder/myapp.ear</code> or <code>file://localhost//D:/myfolder/myapp.ear</code>.</p> <p>JRockit correctly handles UNC paths in version 1.4.2_04.</p>	8.1 SP3
CR135783	<p><code>WebLogicMBeanMaker</code> yielded different results with JDK version 1.4.1_05 and 1.4.2_03.</p> <p>With JDK 1.4.1_05, when a source folder was specified that contained no class files, <code>WebLogicMBeanMaker</code> silently ignored the error and created an MBean JAR file.</p> <p>With JDK 1.4.2, <code>WebLogicMBeanMaker</code> failed if there were no class files in the source path.</p> <p>Now the <code>WebLogicMBeanMaker</code> behavior is the same for both JDK 1.4.1 and JDK 1.4.2.</p>	8.1 SP3
CR136167	<p>The Administration Console could not display the server log in a Japanese locale because of a problem in JDK 1.4.2_03 and earlier JDK 1.4.2 releases.</p> <p><code>WebLogic Server 8.1SP3</code> includes JDK 1.4.2_04, which includes a fix that prevented displaying the server log in a Japanese locale.</p>	8.1 SP3

## Node Manager

Change Request Number	Description	Release Fixed
CR076968	UnixProcessControlNodeManager.c sends a SIGTERM in offline, when it should	7.0 SP6
CR177156	send a SIGKILL, as JavaProcessControl does.  Following a code change, UnixProcessControl offline now returns SIGKILL.	8.1 SP3
CR120410	When Node Manager was stopped and restarted, sometimes after restart Node Manager detected that it was monitoring a Managed Server. It waited for 180 seconds for the Managed Server to contact it, after which it tried to restart the Managed Server. The Managed Server has a timer which tried repeatedly to contact the Node Manager. The timer interval increased exponentially. It tried after 2,4,8,16... seconds. If the Node Manager was started after a long time then the Managed Server timer interval could be larger than 180 seconds, leading the Node Manager to behave as if the Managed Server was down. Additionally, the Managed Server was not starting a timer when the Node Manager failed because it received a NumberFormatException and the thread ended.  Two changes were implemented.  1) The Managed Server timer interval is restricted to a maximum of 128 seconds which is less than the 180 second interval at which the Node Manager attempts a restart.  2) When the Node Manager failed, the Managed Server got an exception. That exception is caught in NodeManagerCommandListener and a timer thread is started to attempt to connect to the Node Manager periodically.	8.1 SP3
CR127930	Please review the security advisory information at	7.0 SP5
CR135441	<a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_51.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_51.00.jsp</a> .	8.1 SP3

## Operations, Administration, and Management

Change Request Number	Description	Release Fixed
CR099488	WebLogic Server calculated the Administration port URL incorrectly. A code change ensures that the URL is calculated based on the Administration Port if it is enabled or, secondly, on the Administration channel if it is enabled.	7.0 SP6 8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR105572	<p>When using <code>wlconfig</code> Ant task to create a JMS server and two JMS queues for a running server instance, the JMS Server and the two JMS queues were created and reflected correctly through the Administration Console. However, <code>wlconfig</code> did not properly update the <code>config.xml</code> file, because it incorrectly placed the 2nd JMS queue outside the <code>&lt;/JMSServer&gt;</code> end tag.</p> <p>A code fix resolved the problem.</p>	8.1 SP3
CR106122 CR124247	<p>Previous releases of WebLogic Server required specification of the <code>NAME</code> attribute of the <code>MLET</code> element.</p> <p>To comply with JMX specifications, the <code>NAME</code> attribute is now optional.</p>	7.0 SP4 8.1 SP3
CR111199	<p>Restarting the Administration Server and then starting two Managed Servers simultaneously resulted in heavy CPU consumption. One Managed Server completed initialization, the second started initialization and stopped.</p> <p>Use of <code>kill -QUIT pid</code> revealed that the main thread was stuck in <code>HashMap.rehash</code>, as shown:</p> <pre>"main" prio=5 tid=0x29240 nid=0x1 runnable [0xffbec000..0xffbedbd4] at java.util.HashMap.rehash(HashMap.java:292) at java.util.HashMap.put(HashMap.java:344) at weblogic.management.internal.DynamicMBeanImpl\$XInfo.get(DynamicMBeanImpl.java:2270) at weblogic.management.internal.DynamicMBeanImpl.&lt;init&gt;(DynamicMBeanImpl.java:195) at weblogic.management.internal.DynamicMBeanImpl.&lt;init&gt;(DynamicMBeanImpl.java:167) at weblogic.management.runtime.RuntimeMBeanDelegate.&lt;init&gt;(RuntimeMBeanDelegate.java:70) at weblogic.jms.frontend.FEConsumer.&lt;init&gt;(FEConsumer.java:100) at weblogic.jms.frontend.FESession\$3.run(FESession.java:952) at ....</pre> <p>A code change increased the size of the default <code>HashMap</code> to accommodate all <code>MBean</code> attributes without resizing, and synchronized thread access to the <code>HashMap</code>.</p>	7.0 SP6 8.1 SP3
CR113455 CR132578 CR185306	<p>The <code>weblogic.Admin</code> command returns to the main class. The main class returns an exit code that is determined by whether or not an exception was thrown. An unnecessary exception was thrown when the command succeeded.</p> <p>WebLogic Server no longer throws an exception when a command succeeds.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR120398	<p><code>weblogic.common.internal.AdminProxy</code> code threw an exception if the caller was not an admin user. The operator user requests were dispatched to the default queue. The server hung during shutdown behaving as if it had a pending request on the default queue. But the admin user requests were dispatched to a different queue than the default.</p> <p>Code was added to enable WebLogic Server to check if the user is an operator, and if so, allow the user to shut down the server. Additional code was added to dispatch the operator user requests to a queue other than the default. This resolves the problem.</p>	8.1 SP3
CR120926	<p>Managed server logs could not be viewed when attempting a remote connection to the Managed Server because the url specified the wrong port when the administration port or admin channel were used.</p> <p>WebLogic Server was updated to use the correct port and do the operation with <code>QOS_ADMIN</code> set.</p>	8.1 SP3
CR121015	<p>MBean auditing was recording clear-text passwords in certain cases where properties were tagged with <code>@exclude</code>.</p> <p>A code change ensures that WebLogic Server does not print the values of the encrypted attributes in clear text.</p>	8.1 SP3
CR121728	<p>Enhancement:</p> <p>Mbeans for custom authentication providers no longer must be put inside of the WebLogic installation tree in the <code>WL_HOME\server\lib\mbeantypes</code> directory.</p> <p><code>WL_HOME\server\lib\mbeantypes</code> is the default directory for installing MBean types. However, if you want WebLogic Server to look for MBean types in <i>additional</i> directories, use the <code>-Dweblogic.alternateTypesDirectory=&lt;dir&gt;</code> command-line flag when starting your server, where <code>&lt;dir&gt;</code> is a comma-separated list of directory names. When you use this flag, WebLogic Server will always load MBean types from <code>WL_HOME\server\lib\mbeantypes</code> first, then will look in the additional directories and load all valid archives present in those directories (regardless of their extension). For example, if <code>-Dweblogic.alternateTypesDirectory = dirX,dirY</code>, WebLogic Server will first load MBean types from <code>WL_HOME\server\lib\mbeantypes</code>, then any valid archives present in <code>dirX</code> and <code>dirY</code>.</p> <p>Note that you must continue to use the option or your server will be come unbootable (for example, if you used the option and created some users and then decided not to use the <code>alternateTypesDirectory</code> option).</p> <p>For more information, see <a href="#">“Install the MBean Type Into the WebLogic Server Environment”</a> in <i>Developing Security Providers for WebLogic Server</i>.</p>	7.0 SP5
		8.1 SP3

Change Request Number	Description	Release Fixed
CR121856 CR132551 CR179078 CR183921	<p>Time formatted log files caused infinite loop, resulting in the following issues:</p> <ul style="list-style-type: none"> <li>• During startup, the next rotation time was not set.</li> <li>• More than one log file was created at startup.</li> <li>• The log files were rotated twice because WebLogic Server tried to log an exception to a log file when the log file was closed. During rotation, WebLogic Server tried to write to a closed file, which threw an exception.</li> <li>• Even when the log files were restricted, WebLogic Server would let them grow beyond the file count specified. There was no code in place to delete the older log files.</li> </ul> <p>The correct rotation time is now set during startup.</p> <p>If an exception is thrown during rotation, WebLogic Server no longer logs to the file, but prints it to stdout instead.</p> <p>After each rotation, WebLogic Server now checks to see if the files exceed the limit, if restricted. It now deletes older log files.</p>	8.1 SP3
CR122204	<p>When a Managed Server with a custom COMMO bean was restarted without restarting its Administration Server, the Managed Server could not create an instance of the COMMO bean. This exception occurred:</p> <pre>javax.Management.InstanceAlreadyExistsException is thrown, and the Detailed message in the exception is, The Object name specified 'wlidomain:Name=t, Server=managed1, Type=AppViewRuntime' is not unique across the domain. Please choose an unique Object Name</pre> <p>Analysis revealed that the Mbean was not unregistered on the Administration Server's MBeanServer when it went down. For this reason, upon restart, the Managed Server could not re-create the bean—the Administration Server still had a bean instance with that bean's name in its list of instantiated beans. Mbean names must be unique across a domain.</p> <p>The problem was resolved by a code change to unregister all of a Managed Server's server-specific beans when it goes down. When a Managed Server goes down, the Administration Server receives a <code>PeerGoneEvent</code> and unregisters all MBeans associated with the Managed Server.</p>	7.0 SP5 8.1 SP3
Cr122839	<p>When you set the value of <code>SqlStmtProfilingEnabled</code> through MBeans or manually in the <code>config.xml</code> file and then attempted to retrieve that value through MBeans it did not show the value. Instead, it returned the following message, "It appears that no attributes have been specified for this MBean".</p> <p>Code was added to fix this problem.</p>	8.1 SP3

## Resolved Problems for Service Pack 3

Change Request Number	Description	Release Fixed
CR123821	<p>Could not shut down a server instance using the <code>weblogic.Admin SHUTDOWN</code> command.</p> <p>A code fix ensures that <code>weblogic.admin</code> correctly uses the default URL (that is, <code>t3://localhost:7001</code>).</p>	8.1 SP3
CR124099 CR135050	<p>Applications were deployed on servers to which they were not targeted.</p> <p>Code was added to check the target before deploying applications.</p>	8.1 SP3
CR126529 CR173999	<p>The Weblogic MBeanLoader generated the following error message:</p> <pre>weblogic.management.commo.WebLogicMBeanLoader security.xml Exception in thread "main" weblogic.management.ManagementError: [Management:141113] The management subsystem was accessed prior to its initialization. This was resolved by fixing the weblogicMBeanLoader/weblogicMBeanDumper initialization code.</pre>	8.1 SP3
CR126570 CR132988	<p>In previous service packs, the log method of the <code>java.util.logging.Logger</code> was not honored.</p> <p>A code fix resolved this issue.</p>	8.1 SP3
CR126615	<p><code>-Dweblogic.ProductionModeEnabled=true</code> was not being processed correctly because WebLogic Server was expecting it to be a Boolean rather than a string.</p> <p>This problem was fixed by adding code to make WebLogic Server process this as a string in the ant task.</p>	8.1 SP3
CR126930 CR181748 CR183828	<p>In WebLogic Server 8.1, the log rotation methodology changed.</p> <p>Log rotation was triggered by not only time, but also by the output activity to the log file. In other words, if there was no message at rotation time, WebLogic Server did not rotate the log until it was modified.</p> <p>This problem has been addressed by forcing WebLogic Server to rotate the logs at the scheduled times, after initially rotating based on the first log activity after starting the server.</p>	8.1 SP3
CR127698 CR132901	<p>The Managed Server was started in production mode only when the Administration Server started in production mode. In all other combinations, the Managed Server was started in development mode.</p> <p>Code was added to fix the logic for <code>ProductionModeEnabled</code> on the Managed Server.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR128032 CR174966 CR177166	<p>Users from the Deployers group were not able to deploy applications either from the console or using the command line tool <code>weblogic.Deployer</code>. Deployment of applications required write permissions on certain attributes &amp; execute permissions on other actions.</p> <p>Permissions were added to users in deployers group to be able to deploy application to various targets (clusters, servers, virtual hosts).</p>	8.1 SP3
CR128061 CR177278	<p>When you configured a new Managed Server node using the Administration Console, the execute queue of the new server instance could not be configured on the console until the Administration Server was restarted.</p> <p>Now new server nodes created after the Administration Server has been started have editable default execute queues.</p>	8.1 SP3
CR128506	<p>The Log file was not stored relative to the server directory path.</p> <p>The log file MBean now recognizes where the server was being started (in what directory), so the log can be found and rotated.</p>	8.1 SP3
CR129439	<p>A local context lookup and call for a local server runtime MBean failed.</p> <p>A code change was made so that when server runtime MBeans are called by class type, then "*" domain is used instead of null.</p>	8.1 SP3
CR129539	<p>The <code>weblogic.logging.NonCatalogLogger</code> with the <code>StdoutEnabled</code> flag set to false disregarded this value by displaying the log output to the Administration Console for client-side logging.</p> <p>Now the <code>consoleHandler</code> logger recognizes the <code>stdoutEnabled</code> flag.</p>	8.1 SP3
CR130042	<p>WebLogic Server startup was failing when the configuration file (<code>config.xml</code>) was referencing other XML files.</p> <p>A code fix resolved this problem with JDK 1.4 XML parser. However, other than the fact that the <code>config.xml</code> file is an <code>.xml</code> extension, WebLogic Server does not document the <code>config.xml</code> as an XML-compliant file, nor does the documentation recommend such XML-like references.</p>	8.1 SP3
CR130142	<p>In a clustered configuration, the <code>userExists()</code> call sometimes fails after attempting to create a new user or group on a Managed Server.</p> <p>The problem was resolved by a code fix so that newly created user lookups using the <code>userExists()</code> API occur on the Administration Server, and existing user lookups occur on the local Managed Server.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR130413 CR168268	<p>For Attribute Change Audit Notifications, the auditor printed the old value of the attribute as the new value.</p> <p>A code fix ensures that the auditor uses the correct value as the new value for a given attribute.</p> <p>For detailed information about obtaining auditing information from the Administration Console, see <a href="#">“Auditing Providers”</a> in <i>Developing Security Providers for WebLogic Server</i>.</p>	8.1 SP3
CR130441	<p>In previous version 8.1 service packs, <code>weblogic.Admin</code> did not properly handle the "commotype" syntax. This caused the sample security providers to break. When you built the samples, started the server, and tried to set up the domain, you received a number of <code>javax.management.MBeanException</code> errors.</p> <p>A code change enables <code>weblogic.Admin</code> to correctly handle the "commotype" syntax. As a result, the sample security providers will run correctly against this service pack.</p>	7.0 SP5 8.1 SP3
CR132445	<p><code>weblogic.Admin</code> did not work properly with the default URL. For example: <code>java weblogic.Admin -username ss -password ss VERSION</code> returned no results, but adding <code>-url t3://localhost:7001</code> returned correct results.</p> <p>Following a code change, <code>weblogic.Admin</code> now checks the input URL for null, because it has the default URL internally already.</p>	7.0 SP5 8.1 SP3
CR132824	<p>For Web applications deployed on a cluster, the number of servlets in the Web application displayed in the Administration Console was incorrect because of a <code>java.lang.ClassCastException</code>.</p> <p>The problem was resolved with a code fix.</p>	8.1 SP3
CR133879	<p>The <code>weblogic.properties</code> MBean conversion tool uses a separate MBeanServer than what is used for internal server operations. Therefore, an <code>invalidAttributeValueException</code> was thrown when the MBean being converted was not found in the MBeanServer, which was causing the conversion to fail.</p> <p>A code change was made so that when an MBean properties conversion is in progress, a check is performed to see if the MBean exists in the MBeanServer that the <code>weblogic.properties</code> converter uses before throwing an exception.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR136488	<p>Any file, regardless of extension, is loaded by the plug-in mechanism and produces server start-time errors. Only .jar files in lib/mbeansources should be considered security providers; other suffixed files should be ignored.</p> <p>Previously, any file that was in the .../lib/mbeansources directory would be treated as security providers and be loaded at server boot time. This would happen independently of the file extension type. Now, only files that end in .jar or .zip will be treated as security providers. If you relied on security providers being loaded that were not in a file ending in .jar or .zip, then your domain will not boot any longer as those security providers will no longer be loaded. Those security provider files will have to be renamed to work correctly before booting the server.</p>	8.1 SP3
CR137144	<p>A <code>LogRotationException</code> was thrown when rotating the domain log. This occurred when the logger generated log index(0001) based in a new log directory was rotated into the current log directory where a file by the same name existed.</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3
CR161933 CR177137	<p>The following weblogic.Server command-line options were restored to provide interoperability with previous releases:</p> <ul style="list-style-type: none"> <li>-Dweblogic.management.rmiQueueSize</li> <li>-Dweblogic.management.htmlQueueSize</li> </ul>	8.1 SP3
CR172130	<p>Failure to specify a name while doing an MBean lookup resulted in a Null Pointer Exception.</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3
CR172534 CR174686	<p>WebLogic Server performance has been enhanced by minimizing the synchronization block to reduce excessive thread waiting.</p>	8.1 SP3
CR172579	<p>When the startup class used the JDK1.4 logging API <code>LogManager.getLogManager().readConfiguration()</code>, WebLogic Server stopped the stdout and no further logging messages were available.</p> <p>WebLogic Server now uses an anonymous logger to resolve this issue.</p>	8.1 SP3
CR172872	<p>When a clustered server node was shut down, it was not removed from the migration managers active servers list. Therefore, in situations where a migratable server was then deleted and recreated, it would generate an <code>InstanceNotFoundException</code> (INFE).</p> <p>Now during shutdown of a migratable server, INFEs are ignored when traversing the active server list in the migration manager.</p>	8.1 SP3

Resolved Problems for Service Pack 3

Change Request Number	Description	Release Fixed
CR173623	<p>During startup, the administration server failed to discover its managed servers.</p> <p>A code fix resolved this issue.</p>	8.1 SP3
CR175233	<p>During server shutdown, methods related to the managed server rediscovery process were incorrectly invoked. Also, the deployment callback handler used stale configuration MBean references.</p> <p>This problem was resolved so that Managed Server rediscovery methods are no longer invoked during server shutdown. And the callback handler now uses administration MBean references to avoid stale configuration MBean references.</p>	8.1 SP3
CR177510	<p>Adding or redeploying a module that was part of a large EAR file (600MB) took a very long time.</p> <p>A code change to the WebApp module improved performance:</p> <ul style="list-style-type: none"> <li>• The WebApp module now considers the update of a module when the delta is for changing the whole module.</li> <li>• To reduce the startup time for large application deployment, deployment was modified to persist the staged targets attribute for configured applications.</li> <li>• Remote call traffic between the Administration Server and a Managed Server was reduced during application deployment.</li> </ul>	8.1 SP3
CR180117	<p>The logging handler (<code>wLErrorManager.error</code>) was causing re-entry into the logging code when reporting a fatal error, which was resulting in a logging deadlock with another logging thread.</p> <p>A code change prevents a logging deadlock when logging handlers report a fatal error.</p>	8.1 SP3
CR186194	<p>Certain <code>userLockoutManager</code> operations were incorrectly executed on the Administration server rather than on the local managed server;</p>	7.0 SP5
CR183447		8.1 SP3
CR125018	<p>Code was changed to ensure the operations are executed on the correct server.</p>	

# Plug-Ins

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR084303	WebLogic Server proxy plug-ins restricted the HTTP commands that could be submitted from the client to the server. The validation rules in the plug-in code now allow the following HTTP commands that are needed for WebDAV implementations:  DELETE  GET  HEAD  OPTIONS  POST  PUT  *COPY  LOCK  MKCOL  MOVE  PROPFIND  PROPPATCH  SEARCH  UNLOCK	6.1 SP7  7.0 SP5  8.1 SP3
CR105123	(ISAPI) <code>DefaultFileName</code> did not work if <code>iisforward.dll</code> was not used. The problem was exhibited when: <ul style="list-style-type: none"><li>• Virtual Directory was configured.</li><li>• Mime type was configured to * (proxy everything).</li><li>• <code>DefaultFileName</code> was added to <code>iisproxy.ini</code></li></ul> On a request for a directory that had no filename, the <code>DefaultFileName</code> was not used. The problem was corrected with a code fix.	6.1 SP7  8.1 SP3

Change Request Number	Description	Release Fixed
CR105173	<p>(NSAPI) When a client stopped a response from being sent to it (for example, closing the browser before the response is completely received), a 500 [WRITE TOCLIENT ERROR] was logged in the Web server logs.</p> <p>The Web server health monitoring tools used a 500 error to indicate that something was wrong with the server's health and that since this was not a server health issue but the client terminating the response, the error if any should not have been 500.</p> <p>The request response path is as follows:</p> <pre>client -&gt; proxyWebserver-&gt;plug-in-&gt;wls</pre> <p>and the expected response path is</p> <pre>client &lt;- proxyWebserver&lt;-plug-in&lt;-wls</pre> <p>After WebLogic Server had successfully sent the response, but the Web server had not completely sent it to the client, the client aborted the communication and a 500 error was logged in the Web server's access.log which normally indicates that something is wrong with the server.</p> <p>A code change was implemented so that 500 errors are not generated if the client breaks the connection.</p>	6.1 SP7 8.1 SP3
CR113033	<p>(ISAPI) In WebLogic Server 6.1 Service Pack 4, the plug-in did not recognize the WLTempDir flag for the _wl_proxy folder. The code was fixed to use the flag.</p>	6.1 SP7 8.1 SP3
CR113093	<p>[Apache] When using multiple MatchExpression parameters in httpd.conf to route requests to different locations, as in:</p> <pre>MatchExpression *.jsp WebLogicHost=localhost WebLogicPort=8001  MatchExpression *.html WebLogicCluster=localhost:8001,localhost:8003</pre> <p>Each request overwrote the same global parameter info, which caused requests to go to the wrong location. In the above example, this problem resulted in *.jsp requests going to the server at port 8003.</p> <p>The code was fixed to ensure that each request uses its own copy of the parameter information.</p>	6.1 SP7 8.1 SP3
CR122207	<p>(NSAPI) If KeepAliveEnabled and DynamicServerList were both enabled, the plug-in could leave sockets in a CLOSE_WAIT state. This problem was solved with a code fix.</p>	6.1 SP6 8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR122754	(ISAPI) In WebLogic Server Service Pack 4, the plug-in parameter <code>WLExcludeByPathOrMimeType</code> did not work when forwarding by mime type. This problem was solved with a code fix.	6.1 SP6 8.1 SP3
CR122755	(ISAPI) In an earlier WebLogic Server Service Pack, the plug-in filter was bypassed if ".wforward" was manually appended to a URL. The code was modified to throw a 404 error if the initial request has a mime type of .wforward.	6.1 SP6 8.1 SP3
CR123120 CR123775	(Apache, NSAPI) If the POST method was used through the plug-in and the Content-Length was not defined, the proxy log file would contain message such as:  POST and PUT requests *must* contain a Content-Length.  The code was modified to set a content length of zero (0) if Content-Length is undefined.	6.1 SP6 8.1 SP3
CR124433	(ISAPI) If IIS was configured with <code>WForwardPath=/</code> , the plug-in would try to forward requests even if the server was down. The error page was never served to clients. The plug-in was modified to properly exclude paths in this situation.	6.1 SP6 8.1 SP3
CR124464	(NSAPI) A memory leak was detected that could cause the plug-in to crash. The problem occurred because the plug-in accessed an exception object after the object had been deleted. The code was fixed to retrieve the exception code from the exception object and then delete the object, which prevents the memory leak from occurring.	6.1 SP6 8.1 SP3
CR125690	(ISAPI) In a configuration that included nine IIS servers and nine clustered WebLogic Server instances, IIS crashed every a few hours, writing an Event 37 to the event log. The wiproxy log contained this message:  Thu Oct 09 13:01:46 2003 *****Exception type [WRITE_ERROR_TO_CLIENT] raised at line 1269 of .iisproxy.cpp  The Reader::fill() method was not allocating enough memory while growing the initial buffer. The four bytes used to mark the end of buffer were getting lost which resulted in the core dump. The problem was solved with a code fix.	6.1 SP6 8.1 SP3

## Resolved Problems for Service Pack 3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR126103	<p>(NSAPI) During load testing, when NSAPI was running on HP11.00 proxying to a six-node cluster on two Solaris boxes (three WebLogic Server instances on each), memory consumption steadily increased, and after approximately 50 minutes, the ns-httpd process crashed. The same load test did not crash on HP11.00 or Solaris. Codes in proxy.cpp used <code>strdup()</code>, a native system call. <code>strdup()</code> allocated system memory to the program's heap space. WebLogic Server used Iplanet's <code>FREE</code> macro to free previous allotted space when it is no longer needed. Because <code>FREE</code> did not free the allocated space by <code>strdup()</code> call, the memory leak occurred.</p> <p>The problem was solved by replacing all native <code>strdup()</code> system calls in <code>proxy.cpp</code> with Iplanet's <code>STRDUP</code> macro so the <code>FREE</code> macro knows what to free.</p>	6.1 SP6 8.1 SP3
CR126568	<p>(NSAPI) Plug-in did not handle <code>%0A</code> in the post request gracefully</p> <p>A POST request <code>%0A</code> at the end sent to WebLogic Server through the NSAPI plug-in added extraneous data into the body stream, and headers appeared at the end of the body. Requests sent directly to WebLogic Server were processed correctly.</p> <p>The problem was corrected by code change to the plug-in to detect and handle HTTP/0.9 responses correctly.</p>	6.1 SP6 8.1 SP3

Change Request Number	Description	Release Fixed
CR126982	<p>(NSAPI) When WLEXcludePathOrMimeType was set, the file types were cut in the request to WebLogic Server, but iplanet failed to serve those files instead.</p> <p>For example, this request for a .jsp that contained a .jpg was made:</p> <pre data-bbox="292 531 1036 638">&lt;Object name="test5" ppath="*/weblogic/*"&gt; Service fn="wl_proxy" WebLogicHost="lorna" WebLogicPort="7001" PathTrim="/weblogic" Debug="ALL" DebugConfigInfo="ON" WLEXcludePathOrMimeType="*.jpg" &lt;/Object&gt;</pre> <p>The request for the .jsp was proxied to WebLogic Server, and the .jsp was displayed without the .jpg. Iplanet failed to server the .jpg. The iplanet access log contained this message:</p> <pre data-bbox="292 722 1123 1229">10.40.4.117 - - [28/Oct/2003:11:45:34 -0500] "GET /weblogic/images/logo_tm_onwt.jpg HTTP/1.1" 500 305 I get the following in wlproxy.log: Tue Oct 28 11:45:35 2003 ===== new request ===== Tue Oct 28 11:45:35 2003 INFO: SSL is not configured Tue Oct 28 11:45:35 2003 URI=[/hello.jsp] Tue Oct 28 11:45:35 2003 attempt #0 out of a max of 5 Tue Oct 28 11:45:35 2003 general list: trying connect to '10.40.4.117'/7001/7001 at line 1224 for '/hello.jsp' Tue Oct 28 11:45:35 2003 INFO: New NON-SSL URL Tue Oct 28 11:45:35 2003 Going to check the general server list Tue Oct 28 11:45:35 2003 WLS info : 10.40.4.117:7001 recycled? 0 Tue Oct 28 11:45:35 2003 Hdrs from Client:[accept]=[*/]*] Tue Oct 28 11:45:35 2003 Hdrs from Client:[accept-language]=[en-us] Tue Oct 28 11:45:35 2003 Hdrs from Client:[accept-encoding]=[gzip, deflate] Tue Oct 28 11:45:35 2003 Hdrs from Client:[user-agent]=[Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)]</pre> <p>WLEXcludePathOrMimeType should cause WebLogic Server not to service the request, and to pass control to the Web server, allowing it to continue processing the request.</p> <p>The problem was solved with a code change.</p>	6.1 SP6 8.1 SP3
CR127231	<p>A request did not fail over to the next available server in the cluster after receiving 503 HTTP status. The same server was tried repeatedly until a READ_ERROR_FROM_SERVER or a CONNECTION_REFUSED exception was raised.</p> <p>Code now marks the server as failed on getting a 503 HTTP status error, gets the next available server and re-sends the request. All requests now successfully fail over to next available server.</p>	7.0 SP5 8.1 SP3

Change Request Number	Description	Release Fixed
CR127658	<p>If a connection was grabbed from the pool, but the server instance has already closed the connection, the <code>HALF_OPEN_SOCKET_RETRY</code> exception was thrown, causing the deletion of the previous connection object and the creation of a new one to connect to the same server.</p> <p>The problem was resolved by the addition of code to handle the <code>HALF_OPEN_SOCKET_RETRY</code> exception properly.</p>	7.0 SP5 8.1 SP3
CR127973	<p>The ISAPI plug-in sometimes failed after a persistent cookie was added to a servlet session. A correction to the cookie parsing code resolved the problem.</p>	7.0 SP5 8.1 SP3
CR128518 CR136974 CR173517 CR173985	<p>(Apache) The plug-in was dropping sessions after WebLogic Server was restarted. The problem occurred because Apache 1.3.x creates multiple child processes to handle incoming requests. Each process maintains its own server list where each server entry is uniquely identified by the JVMID. Weblogic Server provides the JVMID to the process. When a server instance is restarted, it generates a new JVMID, so a request whose cookie contained a new JVMID could not locate the corresponding primary server plug-in if the JVMID was not refreshed in the plug-in.</p> <p>A code fix ensures that if the JVMIDs extracted from a cookie do not match the ones stored in the server list, WebLogic Server refreshes the JVMIDs.</p>	8.1 SP3
CR129026	<p>A memory leak in the ISAPI plug-in was fixed by a code change.</p>	7.0 SP5 8.1 SP3
CR129138	<p>(NSAPI) When the NSAPI plug-in performed name resolution on backend WebLogic Server instances, name resolution used <code>sysGetHostByName</code>, which called <code>getHostByName</code>, which called internal methods that had maximum limits for open file descriptors, causing name resolution to occasionally fail.</p> <p>A fix to cookie parsing and the substitution of JVMIDs to locate primary and secondary servers resolved the problem.</p>	7.0 SP5 8.1 SP3
CR129342	<p>The ISAPI plug-in sent the <code>WL-PATH-TRIM</code> HTTP Header value to a WebLogic Server in place of the <code>WL-PATH-TRIM</code> value.</p> <p>A code change resolved the problem.</p>	7.0 SP5 8.1 SP3
CR129442	<p>Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03_39.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA03_39.00.jsp</a>.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR129471	In previous service packs, the Apache plug-in did not recognize the <code>WLTempDir</code> parameter. This has been corrected.	7.0 SP5 8.1 SP3
CR131640	Request to provide a plug-in for the Sun Java System Web Server (Formerly Sun ONE Web Server).  The Sun Java System Web Server is now fully supported. For more information, see <a href="#">“Supported Web Servers, Browsers, and Firewalls”</a> in the <i>Supported Configurations</i> documentation.	7.0 SP5 8.1 SP3
CR132399	Request for certification of WebLogic Server on HP-UX 11i with Apache 2.0.48.  For more information, see <a href="#">“Supported Web Servers, Browsers, and Firewalls”</a> in the <i>Supported Configurations</i> documentation.	8.1 SP3
CR132690	The WebLogic plug-in was not translating <code>http://hostname:port/</code> to <code>http://hostname:port/index.html</code> when using Netscape 4.0.  A code fix was used to resolve this issue.	8.1 SP3
CR133641	iPlanet users experienced a problem with host name verification and received the following:  INFO: Host () doesn't match (), validation failed ERROR: SSLWrite failed  A code fix resolved this issue.	8.1 SP3
CR134413	(Apache) The plug-in caused a duplicated HTTP header and body for the 302 response. There was no problem between the plug-in and backend servers, but the Apache server added an additional 302 response.  Code was added which reverted the return value of the <code>request_handler</code> method to OK.	7.0 SP5 8.1 SP3
CR135002	In an Apache configuration with multiple virtual hosts, if only one of the virtual hosts was configured with <code>SecureProxy=ON</code> for the WebLogic Server plug-in, and the other virtual hosts did not use <code>SecureProxy</code> or <code>WLProxySSL</code> , the virtual hosts with no SSL configured saw that the plug-in attempted an SSL connection with the backend WebLogic Server. This caused a performance problem.  The problem was resolved with a code change which provides a new parameter that determines whether the SSL connection needs to be initiated.	8.1 SP3
CR135259	The plug-in returned the incorrect status code when the server was down.  A code fix was used to resolve this issue.	8.1 SP3

Resolved Problems for Service Pack 3

Change Request Number	Description	Release Fixed
CR136374	<p>The WebLogic plug-in failed to route requests by path when Apache is configured to use SSL between the browser and Apache.</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3
CR171978	<p>When the <code>FilterPriorityLevel</code> was set in the <code>iisforward.ini</code> file, the forwarding path was broken.</p> <p>A code fix was implemented to ensure that when a virtual host was not defined in the <code>iisforward.ini</code> file, the <code>iisproxy.ini</code> file from the same location as where the <code>iisforward.dll</code> file was loaded is used.</p>	8.1 SP3
CR172072	<p>Provided an enhancement to the <code>WlExcludePathOrMimeType</code> parameter allowing it to be used at the <code>Location</code> tag level.</p> <p><code>WlExcludePathOrMimeType</code> parameter can now be defined locally at the <code>Location</code> tag level as well as globally. When the property is defined locally, it does not override the global property but defines a union of the two parameters.</p>	8.1 SP3
CR172497	<p>(NSAPI) The “<code>pluginparams.ErrorPageTest</code>” failed when attempting a proxy by extension.</p> <p>The workaround for the <code>ErrorPage</code> to be loaded locally is to use the <code>WlExcludePathOrMimeType</code> property if proxying by MIME type.</p> <p>For the SUN One Web Server, version 6.1, some additional configuration steps are required:</p> <ol style="list-style-type: none"> <li>1. Remove <code>NameTrans fn="ntrans-j2ee" name="j2ee"</code> from the default object in the <code>obj.conf</code> file.</li> <li>2. Remove <code>Init fn="load-modules" shlib="C:/Sun/WebServer6.1/bin/https/bin/j2eeplugin.dll" shlib_flags="(global now) "</code> from the <code>magnus.conf</code> file.</li> </ol> <p>For more information, refer to Sun’s documentation, at <a href="http://docs.sun.com/source/817-1831-10/agjava.html#wp1084323">http://docs.sun.com/source/817-1831-10/agjava.html#wp1084323</a></p>	8.1 SP3
CR173581 CR173878	<p>(Apache) The plug-in was logging a confusing “error page is unavailable” log message to the <code>apache_error.log</code>, even when the client had closed the connection.</p> <p>This was resolved by a code change that commented out the erroneous log message.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR173653	<p>When the <code>WLExcludePathOrMimeType</code> property was defined within a <code>Location</code> tag, it should <i>not</i> have had a global scope. (However, when the property was defined outside a <code>Location</code> tag, then it <i>should</i> have had a global scope.)</p> <p>This was resolved by a code change to ensure that the <code>WLExcludePathOrMimeType</code> property is applied only to the requests that match the appropriate <code>Location</code> path for the defined property.</p>	8.1 SP3
CR174431	(NSAPI) The Iplanet plug-in now gracefully handles an EINTR OS error.	8.1 SP3
CR174777	<p>[iPlanet] POST_TIMEOUT errors occurred in the iPlanet log file due to a broken pipe.</p> <p>Code was added to throw a <code>HALF_OPEN_SOCKET_RETRY</code> exception if an <code>EPIPE</code> error is encountered while sending POST data to WebLogic Server.</p>	8.1 SP3
CR175672	<p>(Apache) The Apache server is hanging when the WebLogic plug-in tries to open the <code>wlproxy</code> log file, even though <code>Debug</code> is <code>OFF</code>.</p> <p>The code has been fixed so that the log file is not set if debugging is turned off.</p>	8.1 SP3
CR175989	<p>(Apache) The Apache server generated core dumps when using the worker (multi-threaded) option instead of the <code>prefork</code> (only multi-process) option.</p> <p>This was resolved by fixing the Locking and Unlocking logic.</p>	8.1 SP3
CR177707	<p>When using the release 7.0 SP02 plug-in with client certificates, WebLogic Server worked fine. However, after an upgrade to release 8.1 SP02, the server log reported the following error:</p> <pre>Failed to parse the client certificate in header: WL-Proxy-Client-Cert. Ignoring this certificate. java.security.cert.CertificateException: Could not parse certificate: java.io.EOFException</pre> <p>The error occurred because the 8.1 SP02 plug-in truncated the <code>WL-Proxy-Client-Cert</code> header when it sent it to the server instance.</p> <p>The code was changed so that <code>WL-Proxy-Client-Cert</code> is lazily added to the request sent to WebLogic Server.</p>	8.1 SP3
CR179537	<p>(ISAPI) The IIS proxy plug-in caused heap corruption on the Microsoft Windows platform.</p> <p>The problem was resolved with an internal code fix.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR180236	<p>The release 8.1 SP02 plug-in with client certificates reported the following error:</p> <pre>Failed to parse the client certificate in header: WL-Proxy-Client-Cert. Ignoring this certificate. java.security.cert.CertificateException: Could not parse certificate: java.io.EOFException.</pre> <p>The error occurred because the plug-in truncated the WL-Proxy-Client-Cert header when sending it to the WebLogic Server instance.</p> <p>The problem was resolved with a code fix.</p>	8.1 SP3
CR187184	<p>When using multiple Location tags in a VirtualHost tag, PathPrepend parameter was not working correctly.</p> <p>A code fix was used to ensure that custom properties defined within Location tags are stored locally and do not conflict with global properties.</p>	8.1 SP3

## RMI/RMI-IIOP

Change Request Number	Description	Release Fixed
CR124596	<p>An optional enhancement to the BEA ORB forces reconnection when bootstrapping and allows hardware load-balancers to correctly balance connection attempts.</p> <p>For more information on this feature and its limitations, see <a href="#">Using RMI over IIOP with a Hardware LoadBalancer</a> in <i>Programming WebLogic RMI over IIOP</i>.</p> <p><b>Note:</b> This feature was only partially implemented in WebLogic Server 8.1SP3. See <a href="#">"RMI/IIOP Known Issues"</a> on <a href="#">page 2-23</a> for more details.</p>	8.1 SP3
CR126232	<p>Fragmentation was enabled by default and caused a decrease in performance.</p> <p>Now Fragmentation is disabled by default, which eliminated the decrease in performance.</p>	8.1 SP3
CR128594	<p>WebLogic Server could not handle typecode aliases when reading Java objects on AIX. The code was modified to discard the alias wrapper.</p>	6.1SP6 8.1 SP3
CR129099	<p>IIOP did not explicitly close socket connections on failed sends to non-responsive clients.</p> <p>A code fix resolved the problem.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR132427	<p>A thin client that transmitted large, complex objects concurrently over IIOP using multiple threads would result in the following exception:</p> <pre>Exception in thread ;ExecuteThread: '23' for queue: 'weblogic.kernel.Default'; org.omg.CORBA.MARSHAL: stream corrupted: reading past end of chunk at: 186504 vmcid: 0x0 minor code: 0 completed: No</pre> <p>This was due to GIOP version mismatch in the response triggered by the interleaving of response fragments with out-of-band heartbeat messages. A code fix to place the proper GIOP version in the response resolved this issue.</p>	8.1 SP3
CR136505 CR136542	<p>An IIOP client that started a transaction and executed methods on a CORBA object hosted within the WLS 8.1 ORB was experiencing a rollback exception because the transaction was started in one thread and ended in different thread.</p> <p>A code fix resolved this issue.</p>	8.1 SP3
CR136877 CR175151	<p>The container seemed to cache the RMI Stub incorrectly and raised the issue of ClassCastException casting RMI objects. Once the RMI Stub is created for a Web application, subsequent lookup by another Web application with the same RMI interface as the former returns the Stub for the former Web application and hence causes a ClassCastException. This is due to the incompatibility in classloaders between the Web applications. This problem only occurs when the lookup happens over T3 and HTTP. IIOP does not seem to have the caching implementation in place and hence does not cause the ClassCastException.</p> <p>This problem was fixed with a code change. Two Web applications looking up the same RMI object in the JNDI tree now have their individual stubs instead of trying to use the stub loaded by a previous invocation of one of the Web applications. This is the correct behavior.</p>	8.1 SP3
CR137508	<p>For PeerGone handling in the server, the server pings the client when the muxer makes a request to close down a connection. If this ping hangs, the muxer never shuts down the socket.</p> <p>To fix this, WebLogic Server now sets the failed flag before pinging the client so that a hang counts as a failure.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR176358	<p>While testing <code>ExceptionListener</code> functions in a standalone JMS client with the <code>wlclient.jar</code> and <code>wljmsclient.jar</code> thin clients, the following errors occurred as the connection went down:</p> <pre>Error: CORBA COMM_FAILURE 1398079696 Maybe; nested exception is: org.omg.CORBA.COMM_FAILURE: vmcid: SUN minor code: 208 completed: Maybe</pre> <p>The thin client code was changed so that if the current user on the thread is null, then it will use the anonymous user instead.</p>	8.1 SP3
CR178242	<p>ORBs using an SSL port greater than 32k could not be contacted by WebLogic Server. Such ports were interpreted as a signed short instead of an unsigned short, which resulted in ports greater than 32k being converted to a negative number.</p> <p>The code was modified to treat the port as an unsigned short.</p>	8.1 SP3
CR178243	<p>SSL was not enforceable for IIOP connections established by calling the <code>string_to_object()</code> method on a foreign IOR. This occurred because SSL was only used when there was an SSL port in the SSL tag and bootstrapping occurred over SSL, or when there was no plain port.</p> <p>The code was fixed so that it observes protocol when bootstrapping using ORB functions. As a side effect, it is now possible to force SSL usage when using the <code>string_to_object()</code> method by setting the <code>weblogic.corba.orb.ORBProtocol</code> property to <code>iiops</code>.</p>	8.1 SP3
CR179987	<p>When using the IIOP protocol (for example, when using the WebLogic Server Thin Client or when interoperating with a foreign application server), login attempts using a null password caused a null pointer exception. This failure was not occurring when using the T3 protocol.</p> <p>The code was changed to allow null passwords when using IIOP.</p>	8.1 SP3
CR181321	<p>The WebLogic Thin-client JAAS implementation did not allow passwords over 64 characters.</p> <p>A code fix resolved the problem.</p>	8.1 SP3

# Samples

Change Request Number	Description	Release Fixed
CR130313	<p>The DOCTYPE of the <code>weblogic.xml</code> in the WebLogic Clustering sample had the wrong entity.</p> <p>A space was added between “BEA” and “Systems” in the word “BEASystems”.</p> <p>The final correct, complete DOCTYPE definition should read:</p> <pre>&lt;!DOCTYPE weblogic-web-app PUBLIC "-//BEA Systems, Inc.//DTD Web Application 8.1//EN" "http://www.bea.com/servers/wls810/dtd/weblogic810-web-jar.d td";;&gt;</pre>	8.1SP3
CR132509	<p>When using the .NET C# example included with the MedRec application (<code>%WEBLOGIC_HOME%\samples\server\medrec\src\clients\CSharpClient\bin\ReleaseCSharpClient.exe</code>), the client successfully retrieves a patient record, but when attempting to “Save Changes” from the client application, a date field error was flagged.</p> <p>A code change fixed the date validation.</p>	8.1 SP3

# Security

Change Request Number	Description	Release Fixed
CR090199	<p>Previously, for an <code>init.d</code> script on a UNIX box to gracefully shut down WebLogic Server when the server rebooted, the user name and password needed to be specified on the command-line at invocation of <code>weblogic.Admin</code>.</p> <p>Using a boot identity file plugs this security hole. A new command, <code>StoreUserConfig</code>, has been added to the <code>weblogic.Admin</code> utility and the <code>weblogic.Deployer</code> utility.</p> <p>This command lets you specify the location of a pre-created boot identity (and <code>SerializedSystemIni.dat</code>) file.</p>	7.0 SP5 8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR105933	<p>WebLogic Server was unable to filter connections from LDAP servers because the LDAP protocol was not supported by the connection filter rules.</p> <p>LDAP has now been added to the list of filterable protocols, so that connections from LDAP can be filtered.</p>	<p>7.0 SP5</p> <p>8.1 SP3</p>
CR106192	<p>When you set the security debug flags <code>&lt;ServerDebug DebugSecurityAtn="true" DebugSecurityAtz="true" . . .&gt;</code> with <code>StdoutDebugEnabled="true"</code> <code>StdoutSeverityLevel="64"</code>, the server logged the user password to stdout in clear text.</p> <p>This problem has been resolved by removing the password string from the statement that is written to the log file.</p>	<p>7.0 SP5</p> <p>8.1 SP3</p>
CR107359 CR135441	<p>CR107359 reported a problem in which the Node Manager instantiated its encryption service using the configured private key password as its password. This was an issue when the Node Manager configuration was modified to configure a new private key password because the encryption service could no longer be instantiated.</p> <p>The Node Manager has been modified so that it instantiates its encryption service using an internal string as the password. However, existing encrypted properties cannot be decrypted with the new encryption service, so they must be decrypted with the old service and re-encrypted with the new service.</p> <p>The Node Manager will do this conversion on the first reboot after the service pack is installed (or a patch applied). On any subsequent reboots, the new encryption service is instantiated.</p>	8.1 SP3
CR107373 CR178661	<p>Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_52.01.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_52.01.jsp</a>.</p>	8.1 SP3
CR108624 CR128228	<p>In a performance tuning enhancement, new attributes were added that allow you to limit the depth of a group membership search in an LDAP directory. You can tune the search according to your membership hierarchy, and eliminate searching that you know will not find group members.</p> <p>The new attributes are <code>GroupMembershipSearching</code> and <code>MaxGroupMembershipSearchLevel</code>.</p>	<p>7.0 SP5</p> <p>8.1 SP3</p>
CR112147	<p>The existing <code>ServletAuthentication</code> API methods such as <code>weak</code>, <code>strong</code>, <code>authenticate</code> etc., were not propagating the <code>LoginException</code> back to the caller. Two overridden login methods were added, which perform similarly to the <code>weak</code> and <code>authenticate</code> methods. The <code>assertIdentity</code> method, which performs in the same way as the <code>strong</code> method, was added. These new methods will propagate the <code>LoginException</code> to the calling code.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR112233 CR121114 CR133071 CR135017	Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_61.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_61.00.jsp</a> .	8.1 SP3
CR112471	The All Groups filter was not included in a group search of the embedded LDAP server. Code was added to include the All Groups filter. This filter is enabled through the Administration Console.	8.1 SP3
CR112820 CR112875 CR135819	In the <code>weblogic.security.SSL.TrustManagerJSSE</code> interface, a custom <code>TrustManager</code> contained a setting to govern <code>certificateCallback()</code> behavior. This setting was overridden by variable in <code>validateErr</code> .  A code change resolved this problem. Now SSL in WebLogic Server evaluates results returned by a custom trust manager.	7.0 SP5 8.1 SP3
CR120233 CR123481 CR178414	A code change has resolved the problem that caused a <code>java.lang.ArrayStoreException</code> to be thrown when configuring a custom auditor and a custom role mapper.	8.1 SP3
CR120850	In WebLogic Server 6.1 SP05, <code>weblogic.net.http.HttpsURLConnection</code> did not honor <code>https.nonProxyHosts</code> environment variable. The problem was exhibited in this scenario:  client <--> Proxy <--> Server  The client can be running within WebLogic Server or be a stand-alone Java program. Requests always went through the proxy even if the targeted host was specified in <code>https.nonProxyHosts</code> . If instead, the host is specified in <code>http.nonProxyHosts</code> , the problem does not occur: a direct connection to the host, not to the proxyHost is established, not to the proxyHost, if defined.  Analysis revealed that logic for connecting directly to a host specified in <code>https.nonProxyHosts</code> , even when a proxyHost is defined. This problem did not exist for <code>http.nonProxyHosts</code> , only with <code>https.nonProxyHosts</code> .  Appropriate logic was developed to connected directly to a host specified in <code>https.nonProxyHosts</code> , even when a proxyHost is defined.	6.1 SP6 8.1 SP3
CR121043	Properties set directly in the JSP were not being captured and set in the HTTP client.  A code change has resolved the problem.	7.0 SP5 8.1 SP3

Change Request Number	Description	Release Fixed
CR121310 CR127992 CR155332	Two-way SSL did not working properly with Web Services clients. This problem was resolved by updating the WEBLOGIC_X509_WORKAROUND variation of the <code>convertToCertiom()</code> method to use the WLSDEBUG macros.	8.1 SP3
CR123182	The X509 token used for asserting identity was taken from a certificate not from the request header. This occurred even when X509 identity assertion was not configured.  A test was added to determine if any of the Identity Assertion providers in the security realm were configured to use X509 tokens. If not, the X509 token type is ignored when determining what token to use for asserting identity.	8.1 SP3
CR123716	The security policy for server shutdown was being applied to all server lifecycle methods. The security policy has been updated so that it only applies to server shutdown.	8.1 SP3
CR125592 CR196597	Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04-66.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04-66.00.jsp</a> .	
CR124746 CR175051 CR175045	Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_48.01.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_48.01.jsp</a> .	7.0 SP5 8.1 SP3
CR125911	In a performance tuning enhancement, new attributes were added that allow you to limit the depth of a group membership search. You can tune the search according to your membership hierarchy, and eliminate searching that you know will not find group members.  The new attributes are <code>GroupMembershipSearching</code> and <code>MaxGroupMembershipSearchLevel</code> . For more information see, “ <a href="#">Improving the Performance of WebLogic and LDAP Authentication Providers</a> ” in Managing WebLogic Security.	7.0 SP5 8.1 SP3
CR126062	A <code>NullPointerException</code> was sometimes encountered when data was imported from an LDAP directory.  Code was added to prevent the exception.	8.1 SP3
CR126275	When an Auditing MBean (or a similar non-server MBean) was configured in a domain, the Managed Server tried to add a Notification Listener to the MBean when the server was booting. The Managed Server would then try to cast a <code>NotificationRelay Listener</code> which would cause an exception.  WebLogic Server no longer tries to cast a <code>Notification Relay Listener</code> .	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR126829	<p>The WebLogic Security framework ignored security policies configured in the <code>weblogic-ra.xml</code> deployment descriptor file.</p> <p>This problem was solved by combining the Java security policies and the WebLogic security policies. Security policies specified in the <code>weblogic-ra.xml</code> deployment descriptor file are now honored.</p>	8.1 SP3
CR126837	<p>There was a problem getting a list of users through the <code>listGroupMembers</code> implementation used by the <code>iPlanetAuthenticator</code> MBean. Specifically, the <code>listGroupMembers()</code> method returned an empty cursor.</p> <p>This was resolved by changing the default value of the <code>StaticMemberDNAttribute</code> to “<code>uniquemember</code>”.</p>	8.1 SP3
CR127426	<p>The WebLogic Authentication provider used a dynamic group to represent group membership while the rest of the LDAP Authentication providers used a static group. The recursive search used in the <code>isMember()</code> method of the <code>GroupReader</code> MBean expected a static group; therefore, group membership searches did not work correctly for the WebLogic Authentication provider.</p> <p>Code was added to update the recursive search to handle dynamic groups. The <code>isMember()</code> method of the <code>GroupReader</code> MBean now works consistently for all LDAP-based Authentication providers.</p>	8.1 SP3
CR127964	<p>The embedded LDAP server can become corrupt when an application is processed. The <code>ArrayIndexOutOfBoundsException</code> exception occurs in this situation.</p> <p>Code was added to fix the <code>ArrayIndexOutOfBoundsException</code> exception.</p>	8.1 SP3
CR128054	<p>The signature implementation in the WebLogic Security system incorrectly used the <code>dsig:Id</code> attribute on outbound messages and only checked for <code>dsig:Id</code> on inbound messages.</p> <p>A code change corrected the problem.</p>	8.1 SP3
CR128150 CR130280 CR135329	<p>Users in the “Deployer” global role were prevented from performing the following tasks:</p> <ul style="list-style-type: none"> <li>• Test a JDBC connection pool after it is created (from the Administration Console or from <code>weblogic.Admin TEST_POOL</code>).</li> <li>• Clear the statement cache for a JDBC connection pool.</li> <li>• Suspend, force suspend, destroy, force destroy, or shut down a JDBC connection pool.</li> <li>• Resume or enable a JDBC connection pool.</li> <li>• Deploy or undeploy a J2EE application or application component to a cluster.</li> </ul> <p>Deployers are now permitted to perform these tasks.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR128873	<p>When other JCE providers existed in the classpath, RC4 did not work, which caused the SSL handshake to terminate.</p> <p>Now with the nCiphR JCE provider, the SSL handshake will not break, even though RC4 is being used as the handshake algorithm.</p>	8.1 SP3
CR130467 CR168230 CR172100	<p>If a WebLogic Server domain had two Active Directory Authentication providers configured to use the same Active Directory LDAP server, the connections to the LDAP server would fail if both providers tried to simultaneously access the LDAP server. WebLogic Server would have to be rebooted to clear the bad connection.</p> <p>The Connection Retry Limit attribute was added to the Active Directory Authentication provider. When specified, this attribute to retries connections to the LDAP server if the initial connection failed.</p>	8.1 SP3
CR131535	<p>When a server instance is configured with Verisign SGC/Stepup certificates, the SSL handshake is failing with the following exception:</p> <pre data-bbox="229 881 1002 991">&lt;&lt;WLS Kernel&gt;&gt; &lt;&gt; &lt;BEA-000430&gt; &lt;isMessageComplete javax.net.ssl.SSLProtocolException: FATAL Alert:UNEXPECTED_MESSAGE - A message out of sequence was received.</pre> <p>This has been resolved by adding support for Verisign SGC/Stepup certificates.</p>	8.1 SP3
CR131659	<p>Performance was unduly affected due to synchronization occurring in the <code>SSLSocketFactory.getJSSE</code> call</p> <p>A code fix resolved the problem.</p>	8.1 SP3
CR132276	<p>In a domain with a custom keystore, a user who had the Operator security role was not able to start a Managed Server using the Node Manager. This failure occurred because the Operator security role did not have the required permission to access the secured attributes necessary to start the Managed Server.</p> <p>A code fix ensures that a user assigned the Operator security role can start a Managed Server from the Administration Console using the Node Manager.</p>	8.1 SP3
CR132346 CR174423	<p>The ProxyAuthenticator's constructor not being invoked and not using the <code>SSLLayeredSocket</code> causing a <code>java.net.ConnectException</code>.</p> <p>A change was made to the <code>T3SJvmConnection.newSocket</code> method to use the proxy authenticated socket.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR132949 CR178812	Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_55.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_55.00.jsp</a> .	8.1 SP3
CR133655 CR183238	Connection pool code was not behaving properly, causing the LDAP searches to slow down, which caused, for example, slow authentication with a third-party LDAP. A code change resolved the problem.	7.0 SP5 8.1 SP3
CR134110	The webllogic.Deployer utility threw a <code>java.io.FileNotFoundException</code> when using a STOREUSERCONFIG with username and password and the <code>-upload</code> flag. Code was added to ensure that the deployer uses <code>userconfig</code> in all cases.	7.0 SP6 8.1 SP3
CR134367	The configuration file ( <code>config.xml</code> ) could not be modified when using a <code>setUID</code> script due to a JVM bug in <code>File.canWrite()</code> . A code change workaround has been implemented in WebLogic Server. For more information on the <code>File.canWrite()</code> issue, see: <a href="http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=4993360">http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=4993360</a>	8.1 SP3
CR135488	Previously, any file in the security provider directory was treated as a security provider and loaded at server boot time without regard to its file extension. Now only files that end in <code>.jar</code> or <code>.zip</code> are treated as security providers. Security providers that use other extensions will no longer load. If your domain uses alternate file extensions for security providers, change them for Service Pack 3.	7.0 Sp5, 8.1 SP3
CR136262 CR182644	Embedded LDAP has a search limit = 1000". The problem was resolved with a code change. Embedded LDAP can return more elements when a search query is executed.	8.1 SP3
CR137054	An applet that used the <code>setIgnoreTrustValidation(true)</code> method resulted in a <code>java.lang.ExceptionInInitializerError</code> . A code fix was implemented so that the security manager for an applet would allow an SSL handshake.	8.1 SP3
CR171885 CR099476	Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_59.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04_59.00.jsp</a>	8.1 SP3
CR172151	Authentication errors occur while creating local context after using <code>t3s</code> . This problem was resolved with a code change, which clears the certificate from the thread when the context is closed.	8.1 SP3

Change Request Number	Description	Release Fixed
CR172510	<p>On authentication, the default Authentication provider in the security realm should set the user principal name in the Subject with the same case stored in the LDAP. The Subject was set with whatever case the client uses to log in with (the user name is case-insensitive in the LDAP).</p> <p>This was resolved by adding the <code>UseRetrievedUserNameAsPrincipal</code> attribute to the Details tab on the Administration Console for any WebLogic LDAP Authenticator provider. When enabled, this attribute retrieves the user name from the LDAP to use as the principal name for authentication, instead of the supplied user name.</p>	8.1 SP3
CR172548	<p>The <code>DefaultAuditRecorder.log</code> file in the default domain directory can grow very large and possibly use up the available disk space.</p> <p>This was resolved by providing a command-line argument for specifying a directory where the audit log file will be written:</p> <pre data-bbox="229 826 740 848">-Dweblogic.security.audit.auditLogDir</pre> <p>For more information see “<a href="#">Configuring Security Providers</a>” in <i>Managing WebLogic Security</i>.</p>	8.1 SP3
CR173632	<p>Please review the security advisory information at <a href="http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04-60.00.jsp">http://dev2dev.bea.com/resourcelibrary/advisoriesnotifications/BEA04-60.00.jsp</a>.</p>	8.1 SP3
CR173853	<p>The <code>DefaultAuditRecorder.log</code> file in the default domain directory can grow very large and possibly use up the available disk space.</p> <p>This was resolved by introducing a <code>Rotation Log Minutes</code> attribute, which allows you to specify how many minutes to wait before creating a new <code>DefaultAuditRecorder.log</code> file. At the specified time, the audit file is closed and a new backup file is created.</p> <p>For more information see “<a href="#">Configuring Security Providers</a>” in <i>Managing WebLogic Security</i>.</p>	8.1 SP3
CR174021	<p>The performance when querying Active Directory Group Membership was found to be inefficient.</p> <p>A code fix has resolved this issue.</p> <p>For more information see the “Improving the Performance of WebLogic and LDAP Authentication Providers” section in the “<a href="#">Configuring Security Providers</a>” chapter of <i>Managing WebLogic Security</i>.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR175135	<p>A custom object extending UserInfo could not be passed into InitialContext for authentication, which broke backwards compatibility.</p> <p>The “Custom Object Authentication Enabled” attribute was added to the Security Mbean and the Compatibility Advanced tab on the Administration Console so that if a user passes a custom object that extends UserInfo for authentication, it will be taken care of. This attribute is disabled by default because it affects performance.</p>	8.1 SP3
CR177133	<p>The connection filter for the t3s protocol was not working as documented.</p> <p>A code fix resolved the problem.</p>	8.1 SP3
CR177514	<p>The Administration Console allowed you to create a new user with a password that contained special characters (for example, German umlauts). However, when the new user attempted to log on with that password, the password was not accepted.</p> <p>The problem was resolved with a code that obtains password bytes from an LDAP entry with UTF-8.</p>	8.1 SP3
CR178854	<p>Some of the demonstration certificates and trusted CA certificates shipped in previous services packs of WebLogic Server 8.1 expired on May 14, 2004 or will not work with the Basic Constraints feature. This service pack includes updated trusted CA certificates that will work with the Basic Constraints feature. The certificates (democert.pem, democert1024.pem ca.pem, ca1024.pem, trusted.crt, demo.crt) are provided as files.</p> <p>If you are using a previous service pack of WebLogic Server 8.1, you can upgrade to the updated demonstration certificates and trusted CA certificates following the instructions at <a href="http://dev2dev.bea.com/products/wlserver81/wls_demo_cas.jsp">http://dev2dev.bea.com/products/wlserver81/wls_demo_cas.jsp</a>.</p>	8.1 SP3
CR179066	<p>During server startup, static application deployment was freezing in the LDAP code. This occurred while deploying the security roles for components such as EJBs. The freeze happened because of missed notifications in the LDAP code.</p> <p>This was resolved by a fix to the embedded LDAP code.</p>	8.1 SP3
CR182276	<p>The iPlanet LDAP server returned a wildcard search when performing an explicit user search.</p> <p>A fix was made to the “wild card search” algorithm so that it also matches the “exact name search” algorithm.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR182650	<p>When using identity assertion based on Common Secure Interoperability Version 2 (CSIv2) and Generic Security Service (GSS) tokens, certain lengths of principal names were not properly decoded. This sometimes resulted in the <code>IdentityAssertionProvider</code> not being called, or being called with a truncated principal.</p> <p>A code fix properly encodes and decodes GSS tokens.</p>	8.1 SP3

## Servlets

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR098500	<p>Directory index page could corrupt when the directory had a multi-byte name file and prevent access to files from the corresponding index.</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3
CR108034	Inappropriate error messages generated by a user breaking a connection have been suppressed.	7.0 SP5 8.1 SP3
CR108350	The Administration Console incorrectly indicated that the log file format could be dynamically changed between Common Log Format (CLF) and Extended Log Format (ELF), when such a change actually requires a server reboot. The code was changed to properly indicate the required server reboot when changing this configuration parameter.	7.0 SP5 8.1 SP3
CR108577	Module order was not considered during undeployment.	7.0 SP5
CR175505	This logic is fixed. Now the modules are deactivated and rolled back in the exact reverse order in which they were deployed.	8.1 SP3
CR109885 CR135207	<p>Session replication to a failed secondary node caused the primary node to hang. The primary node appeared to be attempting to connect to the secondary node and holding a lock that other threads had to wait on.</p> <p>A code change has resolved the problem.</p>	7.0 SP5 8.1 SP3
CR110692	The <code>WebAppComponentRuntimeMBean</code> now includes methods that you can use to retrieve information about the cookies that your applications set. For example, you can retrieve the comment and name that identify the cookies your applications set.	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR110798	<p>Calling <code>weblogic.servlet.security.ServletAuthentication.killCookie(req)</code> on a JSP caused the session to remain in WebLogic Server without ever being cleaned up. The code was fixed to ensure that the session is invalidated just before <code>killCookie(req)</code> completes.</p>	7.0 SP5 8.1 SP3
CR120281	<p><code>HttpServletRequest.getParameterValues(String)</code> sometimes returned parameter values twice.</p> <p>Analysis revealed that query parameters in forwarded requests were being checked for parsing in a manner that caused properly parsed parameter values to be returned twice. A code change has resolved the problem.</p>	8.1 SP3
CR120440	<p>When multiple Web Applications were deployed in a Single Sign-On configuration and one application called <code>weblogic.servlet.security.ServletAuthentication.invalidateAll(request)</code>, the <code>HttpSessionListeners</code> in the other applications were not invoked until their session timeouts occurred. This happened because only the session associated with the first Web Applications was registered for invalidation; after the user was authenticated, subsequent sessions were not registered.</p> <p>The code was fixed to ensure that both the session ID and context path of all Web Applications are registered for invalidation as necessary by <code>invalidateAll(request)</code>.</p>	6.1 SP6 8.1 SP3

Change Request Number	Description	Release Fixed
CR121175	<p>HTTPClusterServlet routed requests to a different server instance than the primary instance identified by its JVMID in the session cookie.</p> <p>HTTPClusterServlet was proxying requests to multiple unclustered Managed Servers. The webapp on the backend server instance consisted of an <code>index.jsp</code> and a <code>frameset.jsp</code>. The starting point of the webapp is <code>index.jsp</code>, where a session is created. <code>index.jsp</code> forwards requests using <code>&lt;jsp:forward&gt;</code> to <code>frameset.jsp</code>, which contains a <code>frameset</code>.</p> <p>The first time <code>index.jsp</code> was accessed by a proxy server (HTTPClusterServlet), a session was created and a cookie was set in response. When <code>index.jsp</code> forwarded to <code>frameset.jsp</code>, new requests (for each JSP in <code>frameset</code>) were sent with the cookie. Intermittently, a cookie was found in the request but the request was sent to a server in server list (other the primary server) and the session was lost. The proxy log had the following entry:</p> <pre data-bbox="229 835 1045 968">&lt;Thu Aug 21 15:41:15 PDT 2003&gt;: Found cookie: -1339390245 &lt;Thu Aug 21 15:41:15 PDT 2003&gt;: In-bound headers: &lt;Thu Aug 21 15:41:15 PDT 2003&gt;: Content-Length: 117 &lt;Thu Aug 21 15:41:15 PDT 2003&gt;: ##### Trying to connect with server -1189081773!172.17.26.74!7201!443</pre> <p>The problem was solved with a change to the logic that updates the server list. When updating a JVMID for the server in the list, now the server is removed from list, updated, and then added back to the list. Simply updating the object did not sort the list.</p>	7.0 SP5 8.1 SP3
CR121359	<p><code>Aweblogic.utils.ParsingException</code> occurred when a JSTL end-tag contained a white space between the tag name and the right angle bracket(&gt;).</p> <p>The problem was resolved with a code change to allow zero or one occurrence of white space between an tag's name and '&gt;'.</p>	7.0 SP5 8.1 SP3
CR121846	<p>It was possible for the server to write standard log entries to a log file before the writing Extended Log Format headers. This situation could occur during a log rotation when multiple threads attempted to write to the new log file at the same time.</p> <p>The code was fixed to ensure that the thread handling the log rotation has exclusive access to the new log file until after the log headers are written.</p>	6.1 SP6 8.1 SP3
CR124174	<p>The following setter methods in <code>WebAppComponentMBean</code> have been deprecated: <code>SessionCookieName</code>, <code>SessionCookieComment</code>, <code>SessionCookieDomain</code>, <code>SessionCookiePath</code> and <code>SessionIDLength</code>.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR124600	<p>WebLogic Server sometimes failed to read ELF headers under high load at the initial stage of ELF logging service. This occurred if multiple threads concurrently called the initial stage of ELF logging.</p> <p>A code fix moves the initial method to the constructor rather than calling it lazily.</p>	8.1 SP3
CR125846	<p>According to the Servlet Specification, an exact pattern should take precedence over a wildcard pattern. But this was not working correctly. For example if you have: <code>"/TestPath/*"</code> maps to <code>"WildcardServlet"</code> and <code>"/TestPath"</code> maps to <code>"ExactMatchServlet"</code>, if the incoming relative URI is <code>"/TestPath"</code> then <code>ExactMatchServlet</code> should have been served, not <code>WildcardServlet</code>.</p> <p>This was fixed by making appropriate changes in the pattern matcher.</p>	7.0 SP5 8.1 SP3
CR127050	<p>A <code>NullPointerException</code> sometimes occurred when:</p> <p>A thread <code>t1</code> called <code>getLogStream()</code>, and checked whether the value of <code>logStream</code> field was null or not.</p> <p>Another thread <code>t2</code> tried to rotate log file and changed the <code>logStream</code> value to null.</p> <p>The <code>getLogStream()</code> method returned the <code>logStream</code> value which had been changed by <code>t2</code>.</p> <p>WebLogic Server will now return the <code>logStream</code> value in the synchronized block.</p>	8.1 SP3
CR127621	<p>The value <code>maxKeepAliveSecs</code> for <code>weblogic.management.configuration.WebServerMBean</code> has been made configurable up to a maximum of 300 seconds.</p>	8.1 SP3
CR127644	<p><code>javax.servlet.http.HttpSessionActivationListener</code> and <code>javax.servlet.http.HttpSessionBindingListener</code> are acceptable listener interfaces, but they were missing in <code>WebAppDescriptorComplianceChecker</code> for APPC, so when compiling using <code>wlappc</code>, an error occurs.</p> <p>The missing interfaces were added.</p>	8.1 SP3
CR128051	<p>Clients that post a request that does a forward may see the following warning message:</p> <pre>"Warning: One of the getParameter family of methods called after reading from the ServletInputStream(), can't mix these two!"</pre> <p>A code fix was implemented to ensure that a forwarded request tries to get query parameters only after the post parameters have been parsed to avoid the warning message.</p>	8.1 SP3

Resolved Problems for Service Pack 3

Change Request Number	Description	Release Fixed
CR128234 CR174645	<p>The patch generated by CR121053 caused an NPE when accessing the session ID from the <code>sessionDestroyed()</code> method on the <code>SessionListener</code> and the network channel name is <i>null</i>.</p> <p>The code was changed to check the null value of the network channel.</p> <p>The NPE was occurring because there wasn't an associated request and, therefore no network channel name was returned. Although this was caused by user error, throwing an NPE is not an appropriate response. Therefore, the best solution is to return the default network channel name. However, this will be wrong if network channels are being used.</p>	8.1 SP3
CR128420	<p><code>breakUpAndWriteItOutAsNecessary()</code> tried to separate a manifest header to enforce a maximum of 72 bytes per line, and wrote one line to an output stream at a time. The cause of this problem was that the start offset for the new line is wrong.</p> <p>A code change resolved the problem.</p>	7.0 SP5 8.1 SP3
CR128464	<p>The <code>welcome-file-list</code> entry in the <code>web.xml</code> file was ignored by WebLogic Workshop during iterative development.</p> <p>This was occurring because before the index file could be served, a check was made for its existence. However, because it did not exist in the docroot of the split directory application (which is expected to be the <code>src</code> directory, but which is the <code>outdir</code> directory in the WebLogic Workshop application) it could not be found.</p> <p>The code was changed to look in both the <code>src</code> and <code>outdir</code> directories for index files.</p>	8.1 SP3
CR128624 CR132552	<p>When an application attempted to encode a URL which included an HTML anchor tag, such as this: <a href="http://localhost:7001/WebApp/target.jsp#section4">http://localhost:7001/WebApp/target.jsp#section4</a>, the resulting URL was incorrect: <a href="http://localhost:7001/WebApp/target.jsp#section4;jsessionid=160XDN2vax5g2lbGucG4uspB9h6vzhaZw8KtDLl5urAhK96dqvfo!-1835542719">http://localhost:7001/WebApp/target.jsp#section4;jsessionid=160XDN2vax5g2lbGucG4uspB9h6vzhaZw8KtDLl5urAhK96dqvfo!-1835542719</a>.</p> <p>The anchor tag should be at the end of the URL, so it is ignored. The correct URL would be: <a href="http://localhost:7001/WebApp/target.jsp;jsessionid=160XDN2vax5g2lbGucG4uspB9h6vzhaZw8KtDLl5urAhK96dqvfo!-1835542719#section4">http://localhost:7001/WebApp/target.jsp;jsessionid=160XDN2vax5g2lbGucG4uspB9h6vzhaZw8KtDLl5urAhK96dqvfo!-1835542719#section4</a>.</p> <p>Code was added to look for an anchor if there is no query string in the original URL, and to append the anchor to the end of the encoded URL.</p>	7.0 SP5 8.1 SP3
CR128679	<p>Webapps with <code>save-sessions</code> enabled were leaking memory on undeployment and making bad assumptions about <code>Object.hashCode</code> being unique.</p> <p>A code fix ensures that the webapp context path is used as the key for session data and removes the session data on an undeployment.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR129211	<p>Using <code>ServletOutputStream.write(byte)</code> to write more than the buffer size caused infinite loop.</p> <p>A code change resolved the problem by updating the check for boundary conditions when the buffer is full and <code>autoflush</code> is set to <code>false</code>.</p>	7.0 SP5 8.1 SP3
CR129553	<p>The servlet context listeners were not being invoked in the reverse order of their declarations in the <code>web.xml</code> file during the application (webapp) undeployment.</p> <p>A code change ensures that Servlet context listeners are invoked in the reverse order during undeployment. Therefore, any application that depends on the wrong behavior of context listeners order will be affected by this change, as required by the Servlet specification.</p>	8.1 SP3
CR130051	<p>The following warning message is always logged for memory session data where the session attribute is not serializable and cannot be disabled:</p> <pre>&lt;Warning&gt; &lt;HTTP Session&gt; &lt;BEA-100061&gt; &lt;Web application: ServletContext(id=13577225,name=SessionServlet,context-path= /SessionServlet) tried to place a non-serializable attribute: simplesession.time into the session: 1PNw8xxxxx. This attribute will be lost upon redeployment. This message is logged only once per session.&gt;</pre> <p>This problem was resolved with a code fix so that:</p> <ul style="list-style-type: none"> <li>• The warning message is logged only when the server is running in development mode.</li> <li>• For memory session data, the warning message is logged only when the <code>save-sessions-enabled</code> flag is set to <code>true</code>.</li> </ul>	8.1 SP3
CR130182	<p>When a servlet and a filter used attributes, and if the servlet class was updated after the filter checked whether its <code>ClassLoader</code> had been changed, the <code>ClassCastException</code> could occur because of the difference between the servlet's and filter's classloader.</p> <p>This was resolved with a code change.</p>	8.1 SP3
CR130487	<p>The following error message occurred while running the webauction test due to JSP classes from an older WebLogic Server release being deployed on a release 8.1 server instance:</p> <pre>java.lang.NoSuchMethodError: weblogic.servlet.jsp.StaleChecker.isResourceStale</pre> <p>This is due to changes to the <code>stalechecker</code> interface for release 8.1. This problem was resolved by a code fix so if the <code>isStale()</code> method throws an exception, the exception is caught and the JSPs are regenerated.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR132321	<p>A <i>ClassCastException</i> occurred when using <code>HttpProxyServlet</code> with a <code>WrappedResponse</code> from a <code>Servlet</code> filter.</p> <p>A code fix ensures getting the original response in case of a <code>ResponseWrapper</code>.</p>	8.1 SP3
CR132447	<p>If there is a cookie in a header that contains the <code>SESSIONID</code> name as a substring, the <code>SESSIONID</code> was unexpectedly being changed.</p> <p>This problem was resolved by correcting the logic in parsing <code>SESSIONID</code> from the cookie header.</p> <p>Due to this new change, no new <code>SESSIONID</code> is created when there is an existing <code>SESSIONID</code> or an existing session.</p>	8.1 SP3
CR132488	<p>Editing <code>weblogic.xml</code> from the Administration Console generated a <code>SAXParseException</code>.</p> <p>A code fix changed the order of elements of <code>weblogic.xml</code> according to the <code>weblogic810-web-jar.dtd</code> when generating <code>weblogic.xml</code> through the Administration Console.</p>	8.1 SP3
CR132522	<p>On a Web server without a default Web application, an HTTP request for a missing resource received a response that included an incorrect date header:</p> <pre>HTTP/1.1 404 Not Found Date: Thu, 01 Jan 1970 00:00:00 GMT</pre> <p>This header is not valid according to section 14.18 of RFC2616. A code change resolved the problem.</p>	8.1 SP3
CR133291	<p>A protocol exception, <code>excjava.net.ProtocolException: Didn't meet stated Content-Length</code>, was occurring when a client cancelled a request while the default <code>fileServlet</code> was sending a file.</p> <p>This was resolved with a code change.</p>	8.1 SP3
CR133558	<p><code>ServletContext.setAttribute</code> threw a <code>NullPointerException</code> when a null value was passed to it.</p> <p>The problem was resolved by a code change that calls <code>removeAttribute()</code> when a null value is passed to the <code>setAttribute()</code>.</p>	7.0 SP5 8.1 SP3
CR134412	<p>In a Web application that uses a Struts framework, the classloader and environment context was not set correctly for the application's error pages.</p> <p>A code change was made to set the classloader and environment context correctly for the Web application's error pages.</p>	8.1 SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR134414	<p>When a Serializable Servlet request attribute was added, and then overwritten it with a non-Serializable value, the original value masks the new one.</p> <p>A code change was made to try to remove the value from a HashMap table of serializable attributes if necessary, when replacing a Serializable value with a non-Serializable one.</p>	8.1 SP3
CR134767 CR130216	<p>In development mode, servlet-init was being called twice, first by the Deployment subsystem and second by the App-Poller if the time-stamp of files were changed from its last deployment.</p> <p>A code change prevents GenericPoller from being deployed until WebLogic Server is not in a running state.</p>	8. SP3
CR134813	A code fix was implemented to initialize servlets, context listeners and filters according to the Servlet 2.4 specification.	8.1 SP3
CR136273	<p>When trying to download a file from Internet Explorer, the following error occurred:</p> <p>"Internet Explorer Cannot download a .txt from localhost Internet Explorer was not able to open this internet site. The requested site is either unavailable and cannot be found. Please try again later."</p> <p>This error occurred because the browser failed to handle the situation when both the Content-Disposition and Cache-Control headers were set.</p> <p>Additional code was added to check if both the Content-Disposition and Cache-Control headers were set. If both headers were present, the following warning message is logged to notify users of the situation and help them debug their code :</p> <pre>####&lt;May 4, 2004 7:13:40 PM PDT&gt; &lt;Warning&gt; &lt;HTTP&gt; &lt;mint&gt; &lt;myserver&gt; &lt;ExecuteThread: '13' for queue: 'weblogic.kernel.Default'&gt; &lt;&lt;anonymous&gt;&gt; &lt;&gt; &lt;BEA-1 01324&gt; &lt;Some Browsers may fail when both "Content-Disposition" and "Cache-Control" are set.&gt;</pre>	8.1 SP3

Change Request Number	Description	Release Fixed
CR136735	<p>A SessionData memory leak occurred in the Servlet Container when using Custom Logger and calling getRemoteUser.</p> <p>The reference count of a SessionData may be increased while logging the request if a custom logger calls the HttpAccountingInfo.getRemoteUser(), getRemoteUse(), getRequestedSessionId(), getUserPrincipal() or isRequestedSessionIdValid(), because these methods try to acquire the SessionInternal object. But the reference count is not decremented.</p> <p>The problem was resolved with a code change. The code change caches the values returned from getRemoteUse(), getRequestedSessionId(), getUserPrincipal() and isRequestedSessionIdValid() of ServletRequestImpl into the HttpAccountingInfoImpl object before logging the request, and use the cached value if a custom logger calls the methods of HttpAccountingInfo. This avoids incrementing the reference count.</p>	8.1 SP3
CR137009	<p>The administration console ignored the setting of the wapEnabled attribute.</p> <p>A code fix was implemented so that when wapEnabled was checked by the user and URLRewriting was set to true, the sessionId returned in the URL contains no special characters and has a length of less than 52 characters.</p>	8.1 SP3
CR137162	<p>WebLogic Server was not adding entries for directory listings when it created .war files.</p> <p>A code fix ensures that after a user edits the deployment descriptors for .war files and persists those changes, the new .war file has the proper entries for both the directory and the files contained within it.</p>	7.0 SP5, 8.1 SP3
CR143448	<p>The java.lang.IllegalStateException: HttpSession is invalid exception occurs in the servlet container's internal call. If other threads using the same session ID invalidate the session object during processing of ServletRequestImpl.syncSession(), an IllegalStateException may occur while calling SessionData.putValue() or SessionData.isNew().</p> <p>Ignore the IllegalStateException if the session has been invalidated by other threads.</p>	6.1 SP07 8.1 SP3
CR173782	<p>Some examples in the final version of the JavaServer Faces v1.0 Reference Implementation fails to preload on startup with the following ServletException:</p> <pre data-bbox="229 1407 1065 1515">&lt;Mar 16, 2004 10:31:09 AM MST&gt; &lt;Error&gt; &lt;HTTP&gt; &lt;BEA-101216&gt; &lt;Servlet: "Faces Servlet" failed to preload on startup in Web application: "jsf-guessNumber". javax.servlet.ServletException</pre> <p>The problem was fixed by adding logic during the Web application startup for registering servlet listeners, which could be defined in .tld files in the lib folder of the application.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR176207	<p>In <code>FormBasedAuthentication</code>, the requested URL was not getting properly stored, both before and after the user was being authenticated using the form-based login page.</p> <p>To store the complete URL (including the http or https protocol scheme) of the original request, the following parameter was added to the <code>weblogic.xml</code> file:</p> <pre>&lt;container-descriptor&gt;   &lt;retain-original-url&gt; true &lt;/retain-original-url&gt; &lt;/container-descriptor&gt;</pre>	8.1 SP3
CR176941	<p>The <code>GenericProxyServlet.readStatus</code> fails intermittently. This problem occurs if <code>GenericProxyServlet</code> reuses the connection that had already been closed by a backend server.</p> <p>This was resolved with a code fix to retry the same request when encountering the half-open socket exception and filtering out the connection header.</p>	8.1 SP3
CR177512	<p>If <code>GenericProxyServlet</code> was included within a <code>ResponseWrapper</code> and the content length was equal to zero, then <code>GenericProxyServlet</code> would wait a long time for the return from the <code>java.io.DataInputStream.read()</code> method.</p> <p>A code change ensures that <code>DataInputStream</code> is not read in such cases.</p>	8.1 SP3
CR177521	<p>The <code>ClassPathServlet</code> could not load the class from the <code>Webapp WEB-INF\classes</code> directory. This is because the <code>Webapp</code> module was created with an extra forward slash “/” as a prefix (as mentioned in the <code>context_root</code> of this module in the <code>application.xml</code> file). Therefore, the J2EE application container could not find the appropriate class for this module.</p> <p>The problem was resolved by adding logic to <code>ClassPathServlet</code> so that it considers the extra forward slash of the <code>Webapp</code> module name. Also, the J2EE container was modified to create the <code>Webapp</code> module names for new deployments without using the “/” prefix.</p>	8.1 SP3
CR178684	<p>The <code>servletContext.getRealPath()</code> method was not returning the canonical path.</p> <p>The code was fixed to verify the canonical path in the <code>ServletContext.getRealPath()</code> method.</p>	8.1 SP3
CR129064	<p>Now, when an HTTP request is sent through the iPlanet plug-in, WebLogic Server is no longer incorrectly setting the dynamic server list to WebLogic Cluster 2.</p>	7.0 SP6 8.1 SP3

## SNMP

Change Request Number	Description	Release Fixed
CR109689	When SNMP information was collected using a third-party collector task, the following message was logged: <Error> <SNMP Agent> <000000> < Unable to set Entry Field Value ...> A code fix was implemented to resolve this issue.	8.1 SP3
CR113122	The value that the WebLogic Server SNMP agent returned for <code>sysUpTime</code> did not accurately report the duration since the SNMP agent had been initialized. A code change resolved the problem.	7.0 SP5 8.1 SP3
CR121478 CR136137	Certain token names were longer than the 64 bytes allowed by SNMP. Code was added to make the token names less than 64 bytes.	8.1 SP3
CR135708	The <code>JMSDestinationRuntimeMBean</code> is missing from the list in the types drop down, when creating an SNMP counter monitor. See: SNMP->Traps->Monitors->Counter Monitors->Configure a new Counter Monitor The bean itself is present in the MIB file and can be retrieved with both the MIB explorer and <code>weblogic.Admin GET</code> command. The problem was resolved with a code change which added <code>JMSDestinationRuntime</code> to the dropdown menu for all Monitors (String, Gauge, JMX, and Counter).	8.1 SP3

## Tools

Change Request Number	Description	Release Fixed
CR125017	The <code>wldeploy</code> ant task did not support the <code>failonerror</code> attribute. Setting this attribute generated an error. Code was added to ensure that WebLogic Server now recognizes the <code>failonerror</code> attribute.	8.1 SP3
CR125428	WebLogic Builder was saving the content of the <code>multiplicity</code> attribute in all lower case. A code fix was implemented to comply with Enterprise JavaBeans Specification, Version 2.0.	8.1 SP3

Change Request Number	Description	Release Fixed
CR127407	<p>Using <code>appc</code> without having the <code>weblogic.jar</code> file in system classpath caused the following exception to be thrown:</p> <pre>java.util.MissingResourceException: Can't find bundle for base name weblogic.i18n.J2EELogLocalizer, locale en_US</pre> <p>A code fix allows Ant to implement the correct classloader when the <code>weblogic.jar</code> file is not in system classpath.</p> <p><b>Note:</b> Ant version 1.6 is incompatible with the version bundled in the <code>weblogic.jar</code> file. If you wish to run an external Ant file, you must remove the <code>weblogic.jar</code> file from system classpath.</p>	8.1 SP3
CR127628	<p>No error message was generated for an invalid URL.</p> <p>This problem was corrected by adding code to catch throwable exceptions when creating connections.</p>	8.1 SP3
CR130352	<p>The <code>WLServer</code> Ant task sometimes created a configuration file (<code>config.xml</code>) file without the specified properties. Analysis revealed that <code>WLServer</code> never explicitly called <code>saveDomain</code>, which writes MBean changes to <code>config.xml</code>. Instead, it relied on the trigger calling <code>saveDomain</code>. The problem was that <code>WLServer</code> started and shut down a server so quickly that sometimes the trigger did not happen in time.</p> <p>This problem was resolved by putting a <code>saveDomain</code> call into the <code>WLServer</code> ant task to force the <code>config.xml</code> to be written out before the server is shut down.</p>	7.0 SP5 8.1 SP3
CR131740	<p>There were problems with the <code>wlconfig</code> ANT task's ability to pass multiple values for the properties attribute in a JDBC connection pool configuration.</p> <p>A code fix enables Ant and the WebLogic command-line utilities to set JDBC connection pools that have more than one property.</p>	8.1 SP3
CR134423	<p>The <code>DDInit</code> utility could not create deployment descriptors when an application has large number of EJBs.</p> <p>A code fix ensures that the <code>DDInit</code> utility can open documents with Java encoding strings.</p>	8.1 SP3
CR136468	<p>When a <code>tld</code> file contained multi-byte characters, the header did not match to the file encoding. If editing the tag library using Builder, a <code>tld</code> file is saved by the default encoding. For example, on Japanese Windows, it would be <code>Shift_JIS</code>. However, the header encoding was <code>ISO-8859-1</code> irrespective of file encoding.</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR173675	Running Ant 1.5 and using JDK 1.4.2_0x on Windows 2003 Server resulted in an <code>ntvdm.exe</code> problem, which caused build failures.  This problem was resolved with a code fix.	8.1 SP3

## Web Services

Change Request Number	Description	Release Fixed
CR091230	The WebLogic Server <code>clientgen</code> Ant task was incorrectly allowing hyphens to remain in WSDL files. This caused generated Java code to have class and method names that contained hyphens, which is not legal in java.  A modification to <code>NameUtils</code> now causes <code>clientgen</code> to strip out hyphens. Additionally, for JAXRPC methods and classes, if the resulting string is also a Java keyword, WebLogic Server prepend a <code>_</code> to it, as per JAXRPC Specifications 1.0 and 1.1	7.0 SP5 8.1 SP3
CR102959	When a UDDI operation failed, a "dispositionReport" was returned. The <code>dispositionReport</code> XML was created by using SAAJ classes.  <code>SOAPElement.addChildElement(Name name)</code> now declares the namespace of "name" if its namespace is not already defined.	8.1 SP3
CR103985	Setting the property in " <code>Setting javax.xml.soap.MessageFactory</code> " in the <code>startWebLogic.cmd</code> script did not work properly for <code>org.xml.sax.driver</code> , <code>org.xml.sax.parser</code> , <code>javax.xml.soap.MessageFactory</code> and <code>javax.xml.rpc.ServiceFactory</code> for JSPs and Servlets.  Following a code change, WebLogic Server checks whether the property is already set before setting it, and sets it to the default value if is not set.	7.0 SP5 8.1 SP3
CR105715	The WebLogic Server <code>clientgen</code> Ant task was incorrectly allowing hyphens to remain in WSDL files. This caused generated Java code to have class and method names that contained hyphens, which is not legal in java.  A modification to <code>NameUtils</code> now causes <code>clientgen</code> to strip out hyphens. Additionally, for JAXRPC methods and classes, if the resulting string is also a Java keyword, WebLogic Server prepend a <code>_</code> to it, as per JAXRPC Specifications 1.0 and 1.1	7.0 SP5 8.1 SP3

Change Request Number	Description	Release Fixed
CR106741	<p>Setting useServerTypes=True with multiple service entries did not parse the correct tag, and did not parse more than one tag in web-services.xml. As a result, only the first tag in the web-services.xml was being processed correctly.</p> <p>This was fixed by adding code to look for the correct tag in web-services.xml and process "all". More than one service entry can now be processed for clients setting useServerTypes=True.</p>	8.1 SP3
CR107934	<p>When the encryptKeyName and encryptKeyPass attributes are not specified in the security section for servicegen, the generated web-services.xml file would contain an EncryptionSpec attribute with the value of EncryptBody set to "true". This caused the SOAP messages to be encrypted</p> <p>A code fix resolved this issue by ensuring the web-services.xml file is generated correctly.</p>	8.1 SP3
CR108659	<p>When application resources were not loaded by the same classloader as webserviceclient.jar, an error occurred. Code was added enabling WebLogic Server to correctly handle application resources loaded by multiple classloaders.</p>	8.1 SP3
CR120263	<p>Due to a change in some Security specifications, WebLogic Server has been updated in order to ensure that timestamps are now generated inside the security header.</p>	8.1 SP3
CR120796	<p>Having multiple &lt;soap:operation&gt;s in the same WSDL that are set to different styles is a mixed-style Web Service and is not supported.</p> <p>&lt;soap:operation&gt; now overrides &lt;soap:binding&gt;.</p> <p>If there are multiple operations, each &lt;soap:operation&gt; should be set to the same value. However, if &lt;soap:binding&gt; is set to a value different from &lt;soap:operation&gt;, &lt;soap:operation&gt; will override &lt;soap:binding&gt;.</p>	8.1 SP3
CR121394	<p>Calling a method in the Web service through the ISAPI filter caused this exception:</p>	7.0 SP5
CR128988	<p>java.lang.IllegalArgumentException: Illegal MimeHeader name</p>	8.1 SP3
CR174095	<p>or value</p> <p>A code change has resolved the problem.</p>	
CR121683	<p>&lt;wsr:reliability persistDuration="60000"&gt; entries were generated in WSDL files when reliable messaging was not enabled</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR122033	<p>Images attached to Web services were not working correctly. The problem occurred when the user attached a java.awt.Image to a webservice and the image was improperly serialized/deserialized.</p> <p>Code was added to fix serialization and deserialization.</p>	8.1 SP3
CR122156	<p>The new draft of the x509 profile in the OASIS WSS Spec changed the reference model for x509 certificates.</p> <p>Code was added to change the value type for SKIDs in the KeyIdentifier (to X509SubjectKeyIdentifier) and allow Issuer DN + Serial to be embedded in SecurityTokenReferences.</p>	8.1 SP3
CR122502	<p>In using reliable SOAP Messaging to invoke a Web service from a Sender WebLogic Server instance to a Receiver WebLogic Server instance, the Sender continued to retry invocations beyond the configured retry limit. The retries continued until the Receiver was restarted or the Sender's container killed the client because it exceeded the transaction timeout value. This failure to observe the specified retry limit resulted in long running transactions and hung clients.</p> <p>Code was added to ensure that the sender will not retry beyond the configured retry limit.</p>	8.1 SP3
CR124690	<p>When the simpleType contains attribute final, autotype would fail with the following message: <code>weblogic.xml.schema.model.parser.XSDParseException: invalid attribute "final" in element "xsd:simpleType"</code></p> <p>A code fix was used to allow the attribute final to be used with simpleType.</p>	8.1 SP3
CR124892	<p>Autotype failed to create typemapping for types that derive from anyType:</p> <p>Types that depend on other types that are mapped to SOAPElement (like anyType) have to be mapped to SOAPElement themselves. Previously only the 'hasA' dependency was considered.</p> <p>Code was added to map types that derive from anyType to SOAPElement.</p>	8.1 SP3
CR125082	<p>Enhancement provided the WebLogic XML Digital Signatures API which contains classes to digitally sign and validate SOAP messages.</p> <p>See <a href="http://e-docs.bea.com/wls/docs81/xml/xml_xpath.html#xml_dig_sig_api">Using the WebLogic XML Digital Signatures API at http://e-docs.bea.com/wls/docs81/xml/xml_xpath.html#xml_dig_sig_api</a> for more details.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR125852 CR130095 CR137242	<p>The Apache AXIS client was rejecting a SOAP response message because the elements Message and ErrorCode did not have the same namespace prefix. This occurred because WebLogic Server only qualified the top level element.</p> <p>A code fix introduces a system property, <code>-Dweblogic.xml.schema.binding.qualifytooplevelonly</code>, to determine if the top level element is qualified. Valid values are:</p> <ul style="list-style-type: none"> <li>• <code>true</code>—Qualify the top level element. Default value.</li> <li>• <code>false</code>—Do not qualify the top level element.</li> </ul> <p>Apache AXIX client users should set this value to false.</p>	8.1 SP3
CR126960 CR175031	<p>A NullPointerException occurred while starting the server with a WebLogic Express 8.1 license.</p> <p>Code was added to eliminate this exception.</p>	8.1 SP3
CR127276 CR178573	<p>An exception was thrown when a SOAP message had interlaced SOAP attachments and the start parameter of the MIME header did not match the content ID header.</p> <p>A code fix resolved this issue.</p>	8.1 SP3
CR127344	<p>servicegen failed when the accessors of a Java Bean threw checked exceptions.</p> <p>WebLogic Server will now catch the checked exceptions and re-throw with the runtime exception <code>weblogic.xml.schema.binding.PropertyException</code>.</p>	8.1 SP3
CR127391	<p>The SOAP HTTP Binding states that when an Exception occurs within a WebService, the server should throw a HTTP 500 "Internal Server Error", with a SOAP Fault response representing the exception.</p> <p>Many WebService clients use URLConnection in order to make the HttpURLConnection to the WebService. Weblogic Server overrode the <code>java.net.HttpURLConnection</code> with a different version. WebLogic's version threw an exception from the <code>getInputStream</code> method if the status code was <code>&gt;= 400</code>.</p> <p>Since the exception was thrown there was no way for the WebService client to retrieve the inputstream, and thus it violated the Http SOAP Binding Specification.</p> <p>WebLogic Server now retrieves the inputstream upon a 500 error.</p>	8.1 SP3
CR127396 CR176324 CR184605	<p>Clientgen did not produce the right exception type when an exception extended another exception in a different package.</p> <p>This problem was fixed with a code change.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR127409	<p>WebLogic Server only displayed request/response debugging on the client side.</p> <p>When a browser was used, the debug message went to server out. If a Java client was used, the message should have been seen on the client out.</p> <p>Both client and server debugging are now displayed.</p>	8.1 SP3
CR127610	<p>When generating the stub, the wrong parameter class was used due to typemapping information from the provided types.xml being overwritten by JAX-RPC default.</p> <p>Typemapping information from types.xml is not overwritten anymore. Clients programming against the stub have to use the class that corresponds to the XML element as a parameter (as expected), not the class that corresponds to the type.</p>	8.1 SP3
CR127687	<p>When a WSDL was generated from the JPD and a java client was used to invoke a method, the input request xml from the client was invalid and resulted in a NullPointerException.</p> <p>Code was added to fix this problem.</p>	8.1 SP3
CR128214 CR136606	<p>Could not merge an existing web-services.xml file with a new one using the <code>mergeWithExistingWS</code> attribute in the <code>source2wsdd</code> task.</p> <p>The problem was resolved by a fix to the <code>source2wsdd</code> task when setting <code>mergeWithExistingWS="True"</code>, and by adding a new <code>targetNameSpace</code> attribute to <code>source2wsdd</code>.</p> <p>As a result of this change, you cannot generate a WSDL file when there are two services in the <code>web-services.xml</code>; that is, you cannot specify the <code>wsdlFile</code> attribute for the <code>source2wsdd</code> task, because a WSDL is generated for one service only per <code>web-services.xml</code>.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR128255	<p>When a .Net client (using a 512 bit key to encrypt) invoked a secure WebLogic Web Service (using a 1024 bit key to decrypt) it caused the following exception:</p> <pre> Unhandled Exception: System.Web.Services.Protocols.SoapException: Exception during processing: java.lang.AssertionError: weblogic.xml.stream.XMLStreamException: Unable to decrypt EncryptedKey - with nested exception: [weblogic.xml.security.encryption.EncryptionException: Invalid input length for decryption. Length should be multiple of 128 - Block Size. - with nested exception: [com.rsa.jsafe.JSAFE_InputException: Invalid input length for decryption. Length should be multiple of 128 - Block Size.]] (see Fault Detail for stacktrace)  A change was made to provide a clearer error message to explain mismatched encryption/decryption keys:  Unable to decrypt EncryptedKey: key size of encryption/decryption mismatched </pre>	8.1 SP3
CR128446	<p>The clientgen Ant task failed to generate a client .jar file for a WSDL with multiple services defined, throwing the following error:</p> <pre> Client FAIL Exception during Service Create javax.xml.rpc.JAXRPCException: unable to find port:SubmitStatusRequest This may be because the WSDL file and the generated stub is out sync. Doing clientgen again may fix this problem.  The problem was resolved with a code fix. </pre>	8.1 SP3
CR128747	<p>The service interface generated by wsdl2service did not throw a custom exception when "element" is used in "part". For example:</p> <pre> &lt;message name="WSEException"&gt;   &lt;part element="cio:WSEException" name="WSEException"/&gt; &lt;/message&gt; </pre> <p>However, it worked properly when using "type" instead of "element", as follows:</p> <pre> &lt;message name="WSEException"&gt;   &lt;part name="WSEException" type="cio:WSEExceptionType" /&gt; &lt;/message&gt; </pre> <p>A code change ensures that the wsdl2service generates an exception properly in situations where "element" is used in "part".</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR128771	<p>A security configuration exception was thrown when invoking a secure WebLogic Web Service with encryption only.</p> <p>The code was changed so that is no longer necessary to specify a signature key when not signing response.</p>	8.1 SP3
CR129010	<p>When schema files were not in the same directory of a wsdl file, running the clientgen Ant tool would yield the following error:</p> <pre data-bbox="229 644 1067 1402"> [clientgen] Finished Schema2Java parameter validation [clientgen] schemaURL file:/C:/edrive/462245/AaisTransactionManagerService.WSDL [clientgen] java.io.FileNotFoundException: C:\edrive\462245\Ont.xsd (The system cannot find the file specified) [clientgen] at java.io.FileInputStream.open(Native Method) [clientgen] at java.io.FileInputStream.&lt;init&gt;(FileInputStream.java:103) [clientgen] at java.io.FileInputStream.&lt;init&gt;(FileInputStream.java:66) [clientgen] at sun.net.www.protocol.file.FileURLConnection.connect(FileURLC onnection.java:69) [clientgen] at sun.net.www.protocol.file.FileURLConnection.getInputStream(F ileURLConnection.java:156) [clientgen] at weblogic.xml.schema.model.SimpleSchemaResolver.fetchSchemaLo cation(SimpleSchemaResolver.java:120) [clientgen] at weblogic.xml.schema.model.SimpleSchemaResolver.doExternalLoo kup(SimpleSchemaResolver.java:90) [clientgen] at weblogic.xml.schema.model.SimpleSchemaResolver.resolveSchema Location(SimpleSchemaResolver.java:68) [clientgen] at weblogic.xml.schema.model.XSDSchema.resolveInclusion(XSDSche ma.java:533) [clientgen] at weblogic.xml.schema.model.XSDSchema.resolveInclusion(XSDSche ma.java:524) [clientgen] at weblogic.xml.schema.model.XSDSchema.resolveInclude(XSDSche ma.java:489) </pre> <p>In previous service packs, WebLogic Web Services expected a WSDL file and the to-be-included schemas to reside in the same directory. With this change, Web Services now supports relative includes of schemas. For example:</p> <pre data-bbox="229 1520 1067 1541"> a.wsdl ---import---&gt; b/b.xsd ---include---&gt;b/c/c.xsd </pre>	8.1 SP3

Change Request Number	Description	Release Fixed
CR129536	<p>Using the <code>clientgen</code> Ant task with a document-oriented Web Service caused the following exception:</p> <pre>[clientgen] weblogic.xml.schema.model.XSDValidityException: unable to resolve type name</pre> <p>A code fix resolved the problem.</p>	8.1 SP3
CR130300	<p>During a call to a business process from a standalone Java Web Service client, an attempt was made to retrieve the root element from XMLBean, causing a <code>ClassCastException</code> to be thrown. WebLogic Server wrote out <code>xsi:type</code> for anonymous inline types, which caused a <code>ClassCastException</code> in WebLogic Workshop when retrieving the root element from XMLBean.</p> <p>This problem was resolved with a code fix.</p>	8.1 SP3
CR130490	<p>Previous service packs of WebLogic Server 8.1 did not implement the Web Services "InclusiveNamespaces" correctly. According to the Web Services specification, "all visibly used namespaces used in the document must be added in the InclusiveNamespaces PrefixList," as demonstrated here:</p> <pre>&lt;c14n:InclusiveNamespaces xmlns:c14n="http://www.w3.org/2001/10/xml-exc-c14n#" ; PrefixList="n1"&gt;&lt;/c14n:InclusiveNamespaces&gt;</pre> <p>Previous 8.1 service packs put the namespaces as the value of "InclusiveNamespaces" tag instead of being an attribute of "PrefixList." For example:</p> <pre>&lt;c14n:InclusiveNamespaces xmlns:c14n="http://www.w3.org/2001/10/xml-exc-c14n#" ; &gt;n1/c14n:InclusiveNamespaces&gt;</pre> <p>WebLogic Web Services now adheres to the Web Services specification by using the "PrefixList" attribute. Therefore, documents signed by release 8.1 service pack 2 or earlier cannot be verified by service pack 3, and vice versa.</p>	8.1 SP3
CR131753 CR120847	<p>The XML was getting stripped to the next child node in the tree when passing <code>org.w3c.dom.Document</code> to a <code>doc/literal</code> web service.</p> <p>This has been resolved with a code fix.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR132583	<p>Attempts to generate a <code>clientJar</code> for a Web Service that had <code>http-soap12=true</code> was failing with the following exception:</p> <pre>weblogic.webservice.tools.wsdlp.WSDLParseException: transport not supported:http://schemas.xmlsoap.org/soap12/http</pre> <p>In previous service packs, only the SOAP 1.1 HTTP transport in WSDL files was checked when running the <code>clientgen</code> command. A code fix ensure that checks are done for both SOAP 1.1 and SOAP 1.2 HTTP transport in WSDL files when running <code>clientgen</code>.</p>	8.1 SP3
CR132914	<p>The Web Service SSL client left the socket in the <code>CLOSE_WAIT</code> state after execution. The problem was resolved with a code fix to properly close the socket on the client after the execution.</p>	8.1 SP3
CR134014	<p>The <code>wsdl2service</code> Ant task would only generate the interface of the service. By default, it would not generate the implementation class.</p> <p>The code was changed so that the <code>wsdl2Service</code> Ant task generates the interface and the skeleton of the implementation class if "generateImpl" is set to true.</p> <p>For more information, see "<a href="#">Web Service Ant Tasks and Command-Line Utilities</a>" in <i>Programming WebLogic Web Services</i>.</p>	8.1 SP3
CR134913	<p>WebLogic Server fails to validate .NET signed XML documents that contained timestamps. A code change resolved this issue.</p>	8.1 SP3
CR134931	<p>WebLogic Web Services now implements the following final OASIS Standard 1.0 Web Services Security specifications, dated April 6 2004:</p> <ul style="list-style-type: none"> <li>• Web Services Security: SOAP Message Security</li> <li>• Web Services Security: Username Token Profile</li> <li>• Web Services Security: X.509 Token Profile</li> </ul> <p>Service Pack 2 implemented a Working Draft Version 1.0 of the specification, which was the most up-to-date version of the specification at that time.</p> <p>This implementation is not backwards-compatible with previous version 8.1 service packs. For more information on this lack of compatibility, see "<a href="#">What's New in WebLogic Server 8.1 SP3</a>" on page 1-15.</p> <p>For more information on Web Service security, see "<a href="#">Configuring Security</a>" in <i>Programming WebLogic Web Services</i>.</p>	8.1 SP1

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR135317	<p>The webservice ant task did not obey the second level include with autotype ant task.</p> <p>If the first schema imported the second schema and the second schema included another schema, WebLogic Server did not search for this included schema. Thus autotype failed.</p> <p>The problem was resolved with a code change. WebLogic Server now searches for child includes of imported schema.</p>	8.1 SP3
CR136582	<p>A ClassCastException occurred when adding a filter in front of Weblogic 8.1 WebServiceServlet and at the end of the filter, passing a customized HttpServletRequestRequestwrapper instance to the doFilter(req, resp) call.</p> <p>The problem has been resolved with a code change.</p>	8.1 SP3
CR136804	<p>The web-services.xml descriptor contained an extra element of &lt;xsd:documentation&gt;. For example: &lt;xsd:annotation&gt; &lt;xsd:documentation&gt; &lt;xsd:documentation&gt;</p> <p>The the extra &lt;xsd:documentation&gt; tag was removed.</p>	8.1 SP3
CR137520	<p>Using servicegen to generate a webservice on a service method which throws <code>javax.jms.JMSEException</code> caused a <code>weblogic.xml.schema.binding.BindingException</code> exception.</p> <p>A code fix was used to resolve this issue.</p>	8.1 SP3
CR137522	<p>When creating a WSDL file from a <code>web-services.xml</code> file, the order of operations in the <code>web-services.xml</code> file was not preserved in the WSDL.</p> <p>This problem was resolved with a code change, which preserves the order of operations appearing in the WSDL file while generating the <code>web-services.xml</code>.</p>	8.1 SP3
CR172993	<p>For a document-style Web service, the client was failing to catch a user-defined exception because an incorrect namespace prefix was being returned by the service endpoint.</p> <p>A code fix has resolved the problem.</p>	8.1 SP3
CR173969	<p>A web service endpoint could not understand HTTP requests with the “Accept-Charset: UTF-8, UTF-16” included in the request header.</p> <p>A code change solved the problem.</p>	8.1 SP3
CR175093	<p>Setting the <code>-Dweblogic.webservice.i18n.charset=UTF-8</code> property generates a warning message: <code>&lt;Unrecognized property: webservice.i18n.charset.&gt;</code></p> <p>This was resolved by adding this property to management admin to remove this warning message.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR175471	<p>Accessing an external web service via an outbound proxy server (iPlanet Web Proxy Server) returned a SOAP fault. This occurred because using http and https to invoke a remote web service through proxy server with proxy authentication turned on was not allowed.</p> <p>A code change was made to allow http and https tunneling through a proxy server for a web service client.</p>	8.1 SP3
CR178574	<p>The servicegen Ant task failed with the following error when the service method threw a <code>java.sql.SQLException</code>:</p> <pre>weblogic.xml.schema.binding.BindingException: No default constructor was found for class java.lang.StackTraceElement loaded from file:/usr/bean/jdk141_03/jre/lib/rt.jar!/java/lang/StackTraceElement.class. All classes that will be serialized or deserialized must be non-interface, non-abstract classes that provide a public default constructor - with nested exception: [java.lang.NoSuchMethodException: java.lang.StackTraceElement]</pre> <p>A code change resolved the problem.</p>	8.1 SP3
CR179311	<p>A Web Service client did not invoke a foreign Web Service when attempting to send an empty SOAP header element that had no child elements or attributes. For example, <code>&lt;env:Header/&gt;</code></p> <p>The code was fixed so that empty SOAP header elements are no longer sent if they do not contain child elements and attributes.</p>	8.1 SP3

## WebLogic Type 4 JDBC Drivers

Change Request Number	Description	Release Fixed
CR127143 CR128664	<p>XA Driver for DB2</p> <p>When using the XA WebLogic Type 4 JDBC driver for DB2 in a transaction, after executing a prepared statement and committing the transaction, the prepared statement could not be re-used.</p> <p>The problem was corrected with a code fix to the driver.</p>	8.1SP3

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR127310	<p>Drivers for Microsoft SQL Server and Sybase</p> <p>The WebLogic Type 4 JDBC drivers for MS SQL Server and Sybase did not support stored procedure versions with a full stored procedure name such as <code>&lt;stored_proc_name&gt;;&lt;versionNumber&gt;</code>.</p> <p>A code fix to the drivers resolved the problem.</p>	8.1SP3
CR127434	<p>XA Driver for Sybase</p> <p>XARecover failed with the XA WebLogic Type 4 JDBC driver for transactions that were prepared using a different JDBC driver.</p> <p>A code change to the driver corrected the problem.</p>	8.1 SP3
CR127655	<p>XA Driver for Microsoft SQL Server</p> <p>Performance degradation occurred the XA WebLogic Type 4 JDBC driver for MS SQL Server. The problem was corrected with a code fix to the driver.</p>	8.1SP3
CR127834	<p>Drivers for DB2</p> <p>Calling <code>commit()</code> after a batch update caused the database connection to be reset. The problem was corrected with a code fix to the driver.</p>	8.1SP3
CR128654	<p>XA Driver for DB2</p> <p>When creating an XA connection, the driver interpreted an unknown warning message as an error message and prevented the connection creation.</p> <p>The problem was corrected with a code fix to the driver.</p>	8.1SP3
CR128742	<p>Drivers for DB2, Informix, Microsoft SQL Server, and Sybase</p> <p>The following issues were noted with the DATE and TIMESTAMP data types:</p> <ul style="list-style-type: none"> <li>• DB2—Using a TIMESTAMP data type failed with a server protocol error when running with <code>Connection.setAutoCommit(true)</code>.</li> <li>• Informix—Did not recognize the TIMESTAMP data type.</li> <li>• MS SQL Server—Disallowed implicit conversion of data from the DATETIME data type to the TIMESTAMP data type.</li> <li>• Sybase—Did not recognize DATE or TIMESTAMP data types.</li> </ul> <p>The problem was corrected with a code fix to the drivers. The drivers now support the DATE and TIMESTAMP data types.</p>	8.1SP3

Change Request Number	Description	Release Fixed
CR129559	<p>Drivers for Oracle</p> <p>When using the YYYY-MM-DD and YY-MM-DD date formats in the Japanese locale, the TO_DATE function failed with an SQL exception.</p> <p>A code fix to the driver corrected the problem.</p>	8.1SP3
CR132575 CR133146	<p>XA Driver for Microsoft SQL Server</p> <p>With clients using the XA WebLogic Type 4 JDBC driver, the DBMS can not detect a client disconnect. Client disconnects can cause abandoned transactions, and there was no way to provide a transaction timeout for abandoned transactions.</p> <p>A setting was added to the driver to time out abandoned transactions.</p> <p>WebLogic Server was updated so that if you use the XA WebLogic Type 4 JDBC driver for MS SQL Server, XASetTransactionTimeout is automatically set to true internally, to take advantage of the transaction timeout setting on the driver.</p> <p>See <a href="#">“Support for XAResource Transaction Timeout” on page 1-23</a> for more information about the XASetTransactionTimeout attribute.</p>	8.1SP3
CR133508	<p>Drivers for DB2, Informix, Oracle, and Sybase</p> <p>A PreparedStatement with many parameters caused a StackOverflowError.</p> <p>A code change to the drivers corrected the problem.</p>	8.1SP3
CR134410	<p>Drivers for Sybase</p> <p>When callable statement was used multiple times, incorrect values were returned. The problem was corrected with a code fix to the driver.</p>	8.1SP3
CR178619	<p>XA Driver for DB2 and Microsoft SQL Server</p> <p>Drivers return the error code of RMERR (-3) instead of NOTA (-4) which causes transactions to be retried until AbandonTimeoutSeconds instead of the transaction being resolved on recovery.</p> <p>The problem was corrected with a code fix to the drivers.</p>	8.1SP3

# WebLogic Tuxedo Connector

Change Request Number	Description	Release Fixed
CR100556	<p>XOPEN standard data types were supported in Tuxedo, but not in WebLogic Tuxedo Connector.</p> <p>Three new types were added to <code>weblogic.wtc.jatmi</code> that support the XATMI required buffer types. The new types are <code>TypedXOctet</code>, which implements the <code>X_OCTET</code> buffer type and is identical in semantics to a <code>TypedCArray</code>; <code>TypedXCommon</code> and <code>TypedXCType</code>, which implements the <code>X_COMMON</code> and <code>X_C_TYPE</code> buffer types, respectively. These types are identical in semantics and usage to a <code>TypedView</code>.</p>	8.1 SP3
CR125533	<p>When <code>CLASSPATH</code> does not include an EJB JAR file, the invocation of a session bean's service method triggers object replication logic which results in a call to <code>TypedFML._tmpresent</code>. If the <code>CARRAY</code> field is null, the <code>_tmpresent</code> records the field length as 0 (instead of <code>FLDID_SIZE + FLDLEN_SIZE</code>). Ultimately, <code>_tmpostrecv</code> is called and assumes the field length is <code>FLDID_SIZE + FLDLEN_SIZE</code>. Because <code>_tmpresent</code> did not record this information, a negative value was read as field length. Currently the check made for field length is to check if it is 0. This is the reason for the <code>NegativeArraySize</code> exception.</p> <p>Code was added in <code>_tmpostrecv</code> to check the field length and determine whether it is less than or equal to 0.</p>	7.0 SP5 8.1 SP3
CR127636	<p>When running any <code>tBridge FML</code> or <code>Queue</code> example, <code>tBfrom2jms</code> did not recognize <code>quit</code> from <code>tBsend2jms</code> resulting in a timeout failure.</p> <p>This was fixed by changing the code in <code>tBfrom2jms</code> to receive messages from <code>weblogic.jms.Tux2JmsQueue</code>. The <code>quit</code> is now received.</p>	8.1 SP3
CR128909	<p>The <code>viewj</code> compiler parameter of the encoding routine is set to <code>false</code> if the incoming Java string reference is <code>null</code>. When using the <code>-modify_string</code> option, the compiler should add nulls to empty strings.</p> <p>The fix now sets the parameter of the encoding routine to <code>true</code> for null strings allowing nulls to be added to empty strings.</p>	8.1 SP3
CR129485	<p>The <code>xdr_decode_bstring</code> method returned padding bytes with real bytes for <code>CARRAYS</code>.</p> <p>The fix changed <code>xdr_decode_bstring</code> to discard padding bytes.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR134794	<p>The WLS/WTC v7.0sp4 tBridge did not propagate the CorrelationID for CARRAY type messages (ByteMessage) /Qmessage from Tuxedo to WebLogic JMS.</p> <p>Code was added so that when CARRAY messages are received from a Tuxedo queue, it sets the CorrelationID in the JMS message.</p>	8.1 SP3
CR136356	<p>WebLogic Tuxedo Connector FML/VIEW failures occurred with large string types because the current Weblogic Tuxedo Connector did not pad the non-used space for strings in VIEWS.</p> <p>Now WebLogic Tuxedo Connector always sends the fixed length of a string specified in the VIEW file.</p>	8.1 SP3
CR137437	<p>WebLogic Tuxedo Connector <code>tpcall</code> with <code>TPNOTRAN</code> flag set failed, forcing the rollback of the transaction. This occurs because if the <code>tpcall</code> attempt failed, the transaction was marked for rollback irrespective of the flag <code>TPNOTRAN</code> being set.</p> <p>The fix results in the transaction not being marked for rollback if the flag <code>TPNOTRAN</code> is set for <code>tpcall</code>.</p>	8.1 SP3
CR173656	<p>WebLogic Tuxedo Connector Utility routine can throw a null pointer exception when the <code>xdr_encode_string_length</code> method encounters null strings as input.</p> <p>The fix added a check for null input strings.</p>	8.1 SP3
CR180469	<p>WebLogic Tuxedo Connector coordinates a transaction with multiple Tuxedo domains. If the first domain voted <code>ROLLBACK</code> in the first phase of two-phase commit, other Tuxedo domains were not rolled back.</p> <p>Now all Tuxedo branches are rolled back before throwing an <code>XAException</code>.</p>	8.1 SP3

## WLEC

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR121105	<p>WLEC clients were experiencing the following problems with WLEC connection pools:</p> <ul style="list-style-type: none"><li>• Corruption of the ConnectionPool display in the administration console</li><li>• COMM_FAILURES caused by unregistered endpoints</li></ul> <p>A code fix was implemented to resolve these issues and improve the overall performance of WLEC connection pools.</p>	8.1 SP3
CR132509	<p>Clients making calls through WLEC preferentially created new WLEC connections rather than reusing existing connections until the number of connections exceeded the <code>maxpoolsize</code>. Once this threshold was reached, connections were reused.</p> <p>A code fix was used to provide improved system performance and resource utilization when using WLEC connection pools.</p>	8.1 SP3

## XML

<b>Change Request Number</b>	<b>Description</b>	<b>Release Fixed</b>
CR126140	<p>When configuring a custom xml parser in <code>weblogic-application.xml</code>, a <code>NullPointerException</code> sometimes occurred. This problem was fixed with a code addition.</p>	8.1 SP3
CR130177	<p>An null pointer exception was occurring in a document-oriented service when using <code>javax.xml.transform.Source</code> as a parameter or a return value.</p> <p>The problem was resolved with a code fix.</p>	8.1 SP3

Change Request Number	Description	Release Fixed
CR172469	<p>In previous service packs, you could not determine the WebLogic Server XML parser class objects from the JAXP properties file.</p> <p>To define the XML parser class objects in the JAXP properties, add the following entries to the <code>\$java.home/lib/jaxp.properties</code> file:</p> <pre>javax.xml.parsers.DocumentBuilderFactory=weblogic.xml.jaxp.RegistryDocumentBuilderFactory javax.xml.parsers.SAXParserFactory=weblogic.xml.jaxp.RegistrySAXParserFactory javax.xml.transform.TransformerFactory=weblogic.xml.jaxp.RegistrySAXTransformerFactory</pre>	8.1 SP3
CR180943	<p><code>UnsupportedOperationException</code> sometimes occurred while upgrading from 8.1 applications or domains to 8.1 SP2 or SP3. An incorrect error message “This operation requires xqrl.jar” was displayed with this exception. This error message has now been modified to indicate that the query is too complex to be processed.</p> <p><b>Note:</b> This exception may have occurred because the <code>wlxbean.jar</code> file is not in the server CLASSPATH. So, if this exception occurs during the upgrade process, make sure that <code>wlxbean.jar</code> is in the server CLASSPATH.</p>	8.1 SP3