

## **Oracle® Fusion Middleware**

What's New in Oracle WebLogic Server 12.1.3

12c (12.1.3)

**E41902-09**

July 2015

This document describes new features for all components of Oracle WebLogic Server 12.1.3.

Oracle Fusion Middleware What's New in Oracle WebLogic Server 12.1.3, 12c (12.1.3)

E41902-09

Copyright © 2007, 2015, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

---

---

# Contents

<b>Preface</b> .....	v
Documentation Accessibility .....	v
Conventions .....	v
<b>1 Introduction and Roadmap</b>	
1.1 Document Scope and Audience .....	1-1
1.2 Guide to This Document .....	1-1
<b>2 What's New in Oracle WebLogic Server 12.1.3</b>	
2.1 Docker Support .....	2-2
2.2 Oracle Fusion Middleware 12.1.3 Support .....	2-2
2.3 New APIs and Tools for Application Development .....	2-2
2.3.1 Java API for JSON Processing 1.0 (JSR 353) .....	2-2
2.3.2 Java API for RESTful Web Services (JAX-RS) 2.0 .....	2-3
2.3.3 Server-Sent Events .....	2-3
2.3.4 Java API for WebSocket (JSR 356) .....	2-3
2.3.5 Protocol Fallback for WebSocket Messaging .....	2-3
2.3.6 Java Persistence API (JPA) 2.1 .....	2-4
2.3.7 Updated Spring Support .....	2-4
2.3.8 Updated Maven Support .....	2-4
2.3.9 Classloader Analysis Tool Enhancements .....	2-4
2.4 Runtime Improvements .....	2-5
2.4.1 Whole Server Migration with Dynamic and Mixed Clusters .....	2-5
2.4.2 Unicast .....	2-5
2.4.3 Bundling of Oracle Database 12c Driver .....	2-6
2.4.4 JDBC Data Sources .....	2-6
2.4.5 Messaging .....	2-7
2.4.6 JTA .....	2-7
2.4.7 Transaction Recovery Across Data Centers .....	2-8
2.4.8 Optimizations for Exalogic Elastic Cloud Software .....	2-8
2.4.9 WebLogic Tuxedo Connector .....	2-8
2.5 Manageability Improvements .....	2-8
2.5.1 REST Resources for Server Management .....	2-9
2.5.2 Fusion Middleware Control .....	2-9
2.5.3 Global Policy Attachment Support .....	2-9

2.5.4	Security .....	2-10
2.5.5	Logging Enhancements.....	2-10
2.5.6	WebLogic Diagnostic Framework.....	2-11
2.5.7	WLST .....	2-12
2.5.8	Upgrade.....	2-13
2.5.9	New Configuration Features.....	2-13
2.5.10	Deployment Validation Plug-In Support.....	2-13
2.6	Documentation Library Updates.....	2-14
2.7	Standards Support, Supported Configurations, and WebLogic Server Compatibility .	2-14
2.7.1	Standards Support.....	2-14
2.7.2	Supported Configurations.....	2-17
2.7.3	Licensing Information.....	2-18
2.7.4	WebLogic Server Compatibility .....	2-18
2.7.5	Database Interoperability .....	2-18
2.8	Obtaining Information about Patch Set Updates .....	2-20
2.9	Deprecated Functionality (WebLogic Server 12c).....	2-20
2.9.1	WebLogic Full Client.....	2-20
2.9.2	Log4j .....	2-20
2.9.3	LogMBean.ServerLoggingBridgeUserParentLoggersEnabled Attribute .....	2-20
2.9.4	weblogic.Admin.....	2-20
2.9.5	weblogic.management.username and weblogic.management.password.....	2-20
2.9.6	Maven 11x Plug-In Deprecated .....	2-21
2.9.7	JSP Tags for XML Processing.....	2-21
2.9.8	weblogic.websocket and weblogic.websocket.annotation .....	2-21
2.9.9	ActiveCache.....	2-21
2.9.10	Server Argument to WLST Diagnostics Commands.....	2-21
2.9.11	URL Format for REST Management .....	2-22
2.9.12	Node Manager Log File Rotation Properties .....	2-22
2.9.13	Node Manager Properties .....	2-22
2.10	Removed Functionality (WebLogic Server 12c) .....	2-23
2.10.1	Specifying the Logging Implementation in the Administration Console .....	2-23

---

---

# Preface

This preface describes the document accessibility features and conventions used in this guide—*What's New in Oracle WebLogic Server*.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

### Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

## Conventions

The following text conventions are used in this document:

Convention	Meaning
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.



---

---

# Introduction and Roadmap

This chapter describes the contents and audience for this guide—*What's New in Oracle WebLogic Server 12.1.3*.

This chapter includes the following sections:

- [Section 1.1, "Document Scope and Audience"](#)
- [Section 1.2, "Guide to This Document"](#)

## 1.1 Document Scope and Audience

This document describes new features for all components of Oracle WebLogic Server 12c (12.1.3). This document is a resource for system administrators and operators responsible for monitoring and managing a WebLogic Server installation. It is relevant to all phases of a software project, from development through test and production phases.

## 1.2 Guide to This Document

This document is organized as follows:

- This chapter, [Chapter 1, "Introduction and Roadmap"](#), provides a roadmap and describes the audience for this guide.
- [Chapter 2, "What's New in Oracle WebLogic Server 12.1.3"](#) describes the new and changed functionality in this WebLogic Server release.





---

## What's New in Oracle WebLogic Server 12.1.3

This chapter describes the following new and changed functionality in WebLogic Server 12.1.3:

- [Docker Support](#)
- [Oracle Fusion Middleware 12.1.3 Support](#)
- [New APIs and Tools for Application Development](#)
- [Runtime Improvements](#)
- [Manageability Improvements](#)
- [Documentation Library Updates](#)
- [Standards Support, Supported Configurations, and WebLogic Server Compatibility](#)
- [Obtaining Information about Patch Set Updates](#)
- [Deprecated Functionality \(WebLogic Server 12c\)](#)
- [Removed Functionality \(WebLogic Server 12c\)](#)

### Revision History

Date	Description of Changes
May 2015	<ul style="list-style-type: none"> <li>▪ Added a section describing the use of Docker with WebLogic Server. See <a href="#">Section 2.1, "Docker Support"</a>.</li> <li>▪ Added a section summarizing updates made to the WebLogic Server 12.1.3 documentation library. See <a href="#">Section 2.6, "Documentation Library Updates"</a>.</li> <li>▪ Added a section explaining how to get access to information about fixed bugs and patch set updates. See <a href="#">Section 2.8, "Obtaining Information about Patch Set Updates"</a>.</li> </ul>
June 2015	<ul style="list-style-type: none"> <li>▪ Added a section summarizing the functionality that was removed in WebLogic Server 12.1.3. See <a href="#">Section 2.10, "Removed Functionality (WebLogic Server 12c)"</a>.</li> <li>▪ Updated <a href="#">Section 2.6, "Documentation Library Updates"</a> to include information on the latest updates made to the WebLogic Server 12.1.3 documentation library.</li> </ul>
July 2015	<ul style="list-style-type: none"> <li>▪ Updated <a href="#">Section 2.8, "Obtaining Information about Patch Set Updates"</a> to include information on the "Oracle WebLogic Server Patch Set Update 12.1.3.0.4 Fixed Bugs List."</li> </ul>

## 2.1 Docker Support

Oracle WebLogic Server 12.1.3 is now certified to run inside a Docker container. Docker is a Linux-based container technology that enables you to quickly create lightweight clustered and non-clustered WebLogic Server domain configurations on a single host OS or virtual machines, for either development or production environments.

As part of this certification, Oracle provides Docker files and supporting scripts for building images of Oracle WebLogic Server. These images are built as an extension of existing Oracle Linux images. These scripts and build images are available on GitHub at the following location:

<https://github.com/oracle/docker/tree/master/OracleWebLogic>

For more information, see "Getting Started" in *Running Oracle WebLogic Server on Docker*.

## 2.2 Oracle Fusion Middleware 12.1.3 Support

WebLogic Server 12.1.3 is integrated with other Oracle Fusion Middleware 12.1.3 products. Oracle Fusion Middleware is a comprehensive family of products ranging from application development tools and integration solutions to identity management, collaboration, and business intelligence reporting. For more information, see the Oracle Fusion Middleware 12.1.3 documentation library page at <http://docs.oracle.com/middleware/1213/index.html>.

## 2.3 New APIs and Tools for Application Development

WebLogic Server 12.1.3 provides support for selected new or updated APIs from Java EE 7, such as JAX-RS 2.0, Java API for JSON Processing (JSR 353), Java API for WebSocket (JSR 356), and JPA 2.1. These APIs and related capabilities simplify development of server applications accessed by "rich" clients using lightweight web-based protocols such as REST, WebSocket, and Server-Sent Events. Improvements to development tooling and open source support expand developer choices and simplify creation of development environments.

These updates are described in the following sections:

- [Java API for JSON Processing 1.0 \(JSR 353\)](#)
- [Java API for RESTful Web Services \(JAX-RS\) 2.0](#)
- [Server-Sent Events](#)
- [Java API for WebSocket \(JSR 356\)](#)
- [Protocol Fallback for WebSocket Messaging](#)
- [Java Persistence API \(JPA\) 2.1](#)
- [Updated Spring Support](#)
- [Updated Maven Support](#)
- [Classloader Analysis Tool Enhancements](#)

### 2.3.1 Java API for JSON Processing 1.0 (JSR 353)

WebLogic Server 12.1.3 adds support for the Java API for JSON Processing 1.0 (JSR 353). JSON is a data exchange format widely used in web services and other connected

applications. JSR 353 provides an API to parse, transform, and query JSON data using the object model or the streaming model.

For more information about the Java API for JSON Processing, see "Java API for JSON Processing" in *Developing Applications for Oracle WebLogic Server*.

### 2.3.2 Java API for RESTful Web Services (JAX-RS) 2.0

WebLogic Server 12.1.3 adds support of the Jersey 2.5.1 Java API for RESTful Web Services (JAX-RS) 2.0 Reference Implementation (RI) as a pre-built shared library.

For information about how to develop Java EE web services in WebLogic Server that conform to the Representational State Transfer (REST) architectural style using Java API for RESTful Web Services (JAX-RS), see *Developing and Securing RESTful Web Services for Oracle WebLogic Server*.

### 2.3.3 Server-Sent Events

WebLogic Server 12.1.3 adds support for server-sent events through the integration of the Jersey 2.5.1 library. The Jersey 2.5.1 library provides the Reference Implementation (RI) of JSR-399 (JAX-RS 2.0: Java API for RESTful Web Services) and is delivered as a shared library.

Server-sent events enable servers to push data to Web pages over standard HTTP. To use the Jersey 2.5.1 (JAX-RS 2.0 RI) server-sent events implementation, register the library and declare it in the appropriate deployment descriptors.

For more information, see "Using Server-Sent Events in WebLogic Server" in *Developing and Securing RESTful Web Services for Oracle WebLogic Server*.

### 2.3.4 Java API for WebSocket (JSR 356)

WebLogic Server 12.1.3 adds support for the Java API for WebSocket (JSR 356). Support for the Java API for WebSocket enables application developers to use the standard Java EE 7 WebSocket API for connections over the WebSocket protocol.

The proprietary WebLogic Server WebSocket API that was introduced in release 12.1.2 is deprecated but remains supported for backward compatibility.

Although the JSR 356 Java API for WebSocket coexists with the proprietary WebLogic Server WebSocket API, an application cannot contain calls to both APIs. Only one of the APIs can be used in an application.

For more information, including code examples, see "Using the WebSocket Protocol in WebLogic Server" in *Developing Applications for Oracle WebLogic Server*.

### 2.3.5 Protocol Fallback for WebSocket Messaging

WebLogic Server 12.1.3 adds support for protocol fallback for WebSocket messaging.

Protocol fallback provides a mechanism for using an alternative transport for WebSocket messaging when the WebSocket protocol is not supported. Typically the WebSocket protocol is not supported either because the WebSocket object is not available or because WebSocket frames are blocked by a firewall. In this release, the only supported alternative transport is HTTP Long Polling.

Protocol fallback enables application developers to rely on standard programming APIs to perform WebSocket messaging regardless of whether or not the runtime environment supports the WebSocket protocol. For more information, see "Enabling

Protocol Fallback for WebSocket Messaging" in *Developing Applications for Oracle WebLogic Server*.

### 2.3.6 Java Persistence API (JPA) 2.1

WebLogic Server 12.1.3 supports JPA 2.1, which includes support or enhancements for features including Criteria Bulk Update/Delete, stored procedures, JPQL generic function, injectable entity listeners, `TREAT`, converters, DDL generation, and entity graphs. For the complete JPA 2.1 specification, see "JSR-000338 Java Persistence 2.1 (Final Release)" at <http://jcp.org/aboutJava/communityprocess/final/jsr338/index.html>.

For information about enabling JPA 2.1 as the persistence provider in the EJB container, see "Using JPA 2.1 with Oracle TopLink in WebLogic Server" in *Developing Enterprise JavaBeans for Oracle WebLogic Server*.

### 2.3.7 Updated Spring Support

The WebLogic Server/Spring integration features in WebLogic Server 12.1.3 support Spring Framework versions 3.0.x, 3.1.x, and 4.0.x. (Note that Spring Framework version 3.2.x is not supported.)

For more information about Spring Framework features that WebLogic Server supports for use inside Java EE applications, see *Developing and Administering Spring Applications for Oracle WebLogic Server*.

### 2.3.8 Updated Maven Support

Apache Maven is a software tool for building and managing Java-based projects. WebLogic Server provides support for Maven through the provisioning of plug-ins that enable you to perform various operations on WebLogic Server from within a Maven environment.

In WebLogic Server 12.1.3, the following Maven support updates are included:

- Maven plug-in installation and configuration enhancements.
- Improvements in web services integration of `ws-clientgen`, `ws-jwsc`, and `ws-wsdlc` with the Maven life cycle.
- Update to Apache Maven 3.0.5.
- Support for a new `wlst-client` goal, which is a WLST wrapper for executing WLST online commands without requiring a local installation of WebLogic Server.

For more detail about the updated Maven support, see the following documentation:

- *Developing Applications Using Continuous Integration*
- "Using the WebLogic Maven Plug-in" in *Developing Applications for Oracle WebLogic Server*

### 2.3.9 Classloader Analysis Tool Enhancements

Applications may have multiple manifest references to classes that are not directly present in the applications's classpath, but which are being chained into the classpath by manifest references. In some cases, application developers may not be aware that additional classes have been unknowingly pulled into the application's classpath from other JARs, which in turn have manifest references to other JARs.

In WebLogic Server 12.1.3, the Classloader Analysis Tool (CAT) adds the ability to search through an application's or module's classpath to detect and display the underlying chained manifest references.

For information, see "Using the Classloader Analysis Tool (CAT)" in *Developing Applications for Oracle WebLogic Server*.

## 2.4 Runtime Improvements

WebLogic Server 12.1.3 builds on support from prior WebLogic Server versions to improve the reliability, availability, scalability and performance of WebLogic Server applications in clustered environments, when using new Oracle database features, in multi data center architectures, and on Exalogic Systems.

These support improvements are described in the following topics:

- [Whole Server Migration with Dynamic and Mixed Clusters](#)
- [Unicast](#)
- [Bundling of Oracle Database 12c Driver](#)
- [JDBC Data Sources](#)
- [Messaging](#)
- [JTA](#)
- [Transaction Recovery Across Data Centers](#)
- [Optimizations for Exalogic Elastic Cloud Software](#)
- [WebLogic Tuxedo Connector](#)

### 2.4.1 Whole Server Migration with Dynamic and Mixed Clusters

WebLogic Server 12.1.3 adds support for whole server migration with dynamic and mixed clusters. Whole server migration behavior is the same for all clusters, but configuration differs depending on the cluster type.

For a dynamic cluster, you enable or disable automatic whole server migration in the server template. For a mixed cluster, you first enable automatic whole server migration in the server template used by the dynamic server instances in the mixed clusters. Then, you manually enable automatic whole server migration for any of the configured server instances in the cluster, and choose the machines to which you want to migrate if a server instance fails.

For more information, see "Whole Server Migration with Dynamic and Mixed Clusters" in *Administering Clusters for Oracle WebLogic Server*.

### 2.4.2 Unicast

The algorithm used to assign server instances to groups has been changed in WebLogic Server 12.1.3. The new algorithm has been optimized to provide more flexible scaling of running clusters and to better support use cases where Managed Servers are added to WebLogic Server clusters while the clusters are running.

For more information, see "One-to-Many Communication Using Unicast" in *Administering Clusters for Oracle WebLogic Server*.

## 2.4.3 Bundling of Oracle Database 12c Driver

WebLogic Server 12.1.3 includes Oracle 12c JDBC drivers and is certified to operate with the Oracle 12c database. For more information, see the Oracle Fusion Middleware Supported System Configurations page on Oracle Technology Network.

For information about limitations when using Application Continuity, see "Limitations with Application Continuity with Oracle 12c Database" in *Administering JDBC Data Sources for Oracle WebLogic Server*.

## 2.4.4 JDBC Data Sources

This release includes the following new and changed features:

- [Encrypted Connection Properties](#)
- [Connection Labeling Enhancements](#)
- [CountOfTestFailuresTillFlush and CountOfRefreshFailuresTillDisable](#)
- [Oracle Data Base Testing Using SQL ISVALID](#)
- [Oracle Enable JavaNet FastPath](#)
- [Change to securityCacheTimeoutSeconds Parameter](#)
- [Derby Database Driver Version](#)

### 2.4.4.1 Encrypted Connection Properties

As part of a secure configuration, it may be necessary to provide one or more connection property values that should not appear as clear text in the Data Source descriptor file. Update existing Data Source configurations using the `Encrypted Properties` attribute. See "Using Encrypted Connection Properties" in *Administering JDBC Data Sources for Oracle WebLogic Server*.

### 2.4.4.2 Connection Labeling Enhancements

WebLogic Server provides two new connection pool properties, `ConnectionLabelingHighCost` and `HighCostConnectionReuseThreshold`, to allow a connection pool to use brand-new physical connections to serve connection requests from different tenants without incurring re-initialization overhead on other tenant connections already in the pool. See "Using Initialization and Reinitialization Costs to Select Connections" in *Administering JDBC Data Sources for Oracle WebLogic Server*.

WebLogic Server supports callbacks, such as labeling and connection initialization, in EAR or WAR files used by packaged applications. See "Using Connection Labeling with Packaged Applications" in *Administering JDBC Data Sources for Oracle WebLogic Server*.

### 2.4.4.3 CountOfTestFailuresTillFlush and CountOfRefreshFailuresTillDisable

As of this release, the `CountOfTestFailuresTillFlush` and `CountOfRefreshFailuresTillDisable` attributes are disabled when their value is set to 0. In prior releases, the value was 2147483647.

### 2.4.4.4 Oracle Data Base Testing Using SQL ISVALID

A new Test Table name value of `SQL ISVALID` is now supported for Oracle databases. To improve connection testing performance of your Oracle data source, the default setting of the `Test Table Name` attribute of the connection pool is now `SQL ISVALID`.

See "Database Connection Testing Options" in *Administering JDBC Data Sources for Oracle WebLogic Server*.

#### 2.4.4.5 Oracle Enable JavaNet FastPath

The Oracle Enable JavaNet FastPath attribute on the **Domain Structure > Services > Data Sources > Configuration:Oracle** tab of the WebLogic Server Administration Console is no longer supported.

#### 2.4.4.6 Change to securityCacheTimeoutSeconds Parameter

A maximum security cache limit of 10 minutes has been implemented for the `-Dweblogic.jdbc.securityCacheTimeoutSeconds` parameter. In prior releases, there was no limit.

#### 2.4.4.7 Derby Database Driver Version

The Derby Database driver has been updated to version 10.10.1.1.

### 2.4.5 Messaging

In WebLogic Server 12.1.3, JMS includes the following:

- Active GridLink Support for Oracle AQ JMS. For information, see "Configuring an AGL Data Source for AQ JMS" in *Administering JMS Resources for Oracle WebLogic Server*.
- Support for whole server migration (WSM) with dynamic and mixed clusters. For information, see "Simplified JMS Cluster Configuration" in *Administering JMS Resources for Oracle WebLogic Server*.
- To enable JMS .NET clients developed prior to WebLogic Server 12.1.3 to interoperate with WebLogic Server 12.1.3, set `-Dweblogic.protocol.t3.login.replyWithRel10Content=true`. See "Configuring WebLogic Server" in *Developing JMS .NET Client Applications for Oracle WebLogic Server*.
- Certification with Oracle Streams Advanced Queuing (AQ) JMS in Oracle Database 12c. See "Interoperating with Oracle AQ JMS" in *Administering JMS Resources for Oracle WebLogic Server*.

### 2.4.6 JTA

WebLogic Server 12.1.3 includes the following new and changed JTA features:

- [Transactions without TLogs](#)
- [XA Transaction Cluster Affinity](#)

#### 2.4.6.1 Transactions without TLogs

Oracle improves XA transaction performance by providing the option to eliminate the writing of XA transactions to the TLog. XA transaction resources (Determiners) are used during transaction recovery when a TLog is not present. See "XA Transactions without TLogs" in *Developing JTA Applications for Oracle WebLogic Server*.

#### 2.4.6.2 XA Transaction Cluster Affinity

XA transaction affinity improves performance by allowing server instances that are participating in a global transactions to service related requests rather than

load-balancing these requests to other member servers. See "XA Transaction Cluster Affinity" in *Developing JTA Applications for Oracle WebLogic Server*.

## 2.4.7 Transaction Recovery Across Data Centers

WebLogic Server 12.1.3 includes new documentation that describes XA transaction recovery of WebLogic domains spanning physical sites. For documented procedures and approaches, see "Transaction Recovery Spanning Multiple Domains" in *Developing JTA Applications for Oracle WebLogic Server*.

## 2.4.8 Optimizations for Exalogic Elastic Cloud Software

WebLogic Server 12.1.3 includes additional optimizations for Exalogic Elastic Cloud software, including the following:

- A WebLogic Replicated Store is a high performance storage option for Weblogic Messaging Services that require persistence in Oracle Exalogic Elastic Cloud environments. WebLogic Replicated stores depend on Exalogic replicated memory that is managed by a separately administered Daemon Cluster.
- Cooperative Memory Management (CMM) promotes effective memory utilization by allowing WebLogic Server resources to adjust memory utilization in Exalogic Elastic Cloud environments.
- An optimization for JDBC stores to commit a batch of INSERT or DELETE operations with the last operation of the transaction instead of issuing a separate commit call to database server. This feature saves a server round trip per transaction and benefits applications that have many transactions of a small number of operations or small messages.
- TopLink Exalogic optimizations that include: batch writing, increasing the default cache size, and enable other optimizations such as weaving eager relationships. A runtime tuning agent optimizes the use of query batch fetching and dynamic query processing.

For more information, see *Administering WebLogic Server for Oracle Exalogic Elastic Cloud*.

## 2.4.9 WebLogic Tuxedo Connector

When `TightlyCoupledTransactionsEnabled=true`, if a transaction between WebLogic Server and Tuxedo resources use a Active GridLink (AGL) Data Source with GridLink Affinity enabled, then the XA Affinity Policy is automatically used for the transaction.

## 2.5 Manageability Improvements

WebLogic Server 12.1.3 continues to provide new management features which simplify the configuration, monitoring and ongoing management of WebLogic Server domains and applications. These new features are described in the following sections:

- [REST Resources for Server Management](#)
- [Fusion Middleware Control](#)
- [Global Policy Attachment Support](#)
- [Security](#)
- [Logging Enhancements](#)



- [WebLogic Diagnostic Framework](#)
- [WLST](#)
- [Upgrade](#)
- [New Configuration Features](#)
- [Deployment Validation Plug-In Support](#)

### 2.5.1 REST Resources for Server Management

WebLogic Server 12.1.3 includes a set of Representational State Transfer (REST) resources that support the following WebLogic domain management features:

- Life cycle management for servers, deployments, resources
- Deployment operations for applications and libraries
- Data Source configuration
- Access to server and application logs
- Explicit participation in change management

For information about the RESTful management resources provided in this release, see *RESTful Management Interface Reference for Oracle WebLogic Server*.

### 2.5.2 Fusion Middleware Control

WebLogic Server can now be managed through Fusion Middleware Control. Fusion Middleware Control provides management support for all Fusion Middleware components, including WebLogic Server. Use Fusion Middleware Control to manage WebLogic Server when using other Fusion Middleware products in addition to WebLogic Server.

WebLogic Server support includes the following subsets of functionality:

- Manage WebLogic Server clusters, server instances, and domains
- Deploy and redeploy applications and manage application deployments
- Create and configure JDBC data sources
- Manage WebLogic Server messaging (JMS)
- Create and configure users and groups
- Create and configure server templates

For more information, see *Administering Oracle WebLogic Server with Fusion Middleware Control*.

### 2.5.3 Global Policy Attachment Support

WebLogic web services add support for global policy attachment using policy sets in WebLogic Server 12.1.3. For more information, see the following topics in *Securing Web Services and Managing Policies with Oracle Web Services Manager*:

- "Attaching Policies Globally Using Fusion Middleware Control"
- "Attaching Policies Globally Using WLST"

## 2.5.4 Security

WebLogic Server 12.1.3 includes the following new security features:

- [Creating a Custom Identity Keystore for a Network Channel](#)
- [Authentication of Users Not Defined in Identity Store](#)

### 2.5.4.1 Creating a Custom Identity Keystore for a Network Channel

This version of WebLogic Server adds the ability to configure a network channel to use its own custom identity keystore, and other SSL attributes, that are separate from and that override the default keystore and SSL configuration settings for the Managed Server instance on which the channel is configured. This enables you to configure a WebLogic Server instance to use one identity and SSL configuration on one network channel, and another identity and SSL configuration on other channels.

For more information, see "Configuring an Identity Keystore Specific to a Network Channel" in *Administering Security for Oracle WebLogic Server*.

### 2.5.4.2 Authentication of Users Not Defined in Identity Store

In this version of WebLogic Server, the WebLogic Security Service has been enhanced to support the authentication of users who are not defined in the identity store of the security realm, but who instead are created as virtual users who are assigned subject principals based on attributes contained in X.509 client certificate passed in during the SSL handshake.

The WebLogic Identity Assertion provider is not configured by default to authenticate virtual users. However, by customizing this provider's configuration, you can enable this capability in a WebLogic domain.

The following features are provided to support this enhancement:

- The WebLogic Identity Assertion provider is enhanced to optionally accept a client certificate passed in as an X.509 type token and to support virtual users.
- A new interface is provided, `X509SubjectComponentMapper`, which extends the default user name mapper interface so that additional subject components from X.509 certificate attributes can be mapped to subject principals.
- A Virtual User Authentication provider is included to supply the identity assertion virtual user LoginModule.

For more information, see "Authenticating a User Not Defined in the Identity Store" in *Administering Security for Oracle WebLogic Server*.

## 2.5.5 Logging Enhancements

In WebLogic Server 12.1.3, the WebLogic logging services include the following changes:

- The ability to configure `java.util.logging` logger levels. This new feature allows subsystems such as web services to simply use the `java.util.logging` APIs to log messages without having to manage the logging configuration. Configuring the `java.util.logging` logger levels can be managed by the logging system available in the environment.

In addition, subsystem code does not need be aware of the environment in which it is running; for example, standalone WebLogic Server, the Oracle Fusion Middleware Infrastructure environment, and so on.

For more information, see "Configuring java.util.logging Logger Levels" in *Configuring Log Files and Filtering Log Messages for Oracle WebLogic Server*.

- The Server Logging Bridge has been added to the root logger of the `java.util.logging` logger tree when WebLogic Server starts. This eliminates the need to explicitly configure the Server Logging Bridge.

---



---

**Note:** See "Server Logging Bridge" in *Upgrading Oracle WebLogic Server* for important instructions about configuration changes that may be necessary when upgrading a WebLogic domain in which the Server Logging Bridge is configured.

---



---

## 2.5.6 WebLogic Diagnostic Framework

In WebLogic Server 12.1.3, the WebLogic Diagnostic Framework includes the following changes:

- WLST command changes, described in [Section 2.5.7, "WLST"](#).
- WLDF Runtime Control changes

In the prior version of WebLogic Server, when you used the WLDF Runtime Control to deploy a diagnostic system module based on an external resource descriptor, the WLDF Module name was derived from the external resource descriptor.

In this release, if you deploy multiple instances of a diagnostic system module using the WLDF Runtime Control, WLDF now assigns the diagnostic resource name specified in the `createSystemResourceControl` command as the WLDF Module name. This makes it possible to correlate Harvester records and Watch rule notifications to the specific instances of a diagnostic system module that has been deployed.

For more information, see "WLDF Runtime Control" in *Configuring and Using the Diagnostics Framework for Oracle WebLogic Server*.

- If multiple WLDF JDBC archive schemas exist in the same database, you can specify the particular schema to use for accessing JDBC-based archive tables in that database. This provides a way to distinguish among them when accessing the diagnostic archives. For more information, see "Configuring JDBC Resources for WLDF" in *Configuring and Using the Diagnostics Framework for Oracle WebLogic Server*.
- Example of using WLST to configure the Watch and Notification component and Harvester

A set of scripts that demonstrate using WLST to configure the Watch and Notification and Harvester components have been added to the WebLogic Server code examples suite. These scripts are shown and described in the following sections in *Configuring and Using the Diagnostics Framework for Oracle WebLogic Server*:

- "Example: Configuring a Watch and a JMX Notification"
- "Example: Writing a JMXWatchNotificationListener Class"
- "Example: Registering MBeans and Attributes For Harvesting"

If you install the WebLogic Server code examples and start the WebLogic Server samples domain, you can find these examples under the title "Configuring the

Watch and Notification System and Harvesting Data Using WLST" when the samples web page is launched.

For information about installing and configuring the WebLogic Server code examples, see "Sample Applications and Code Examples" in *Understanding Oracle WebLogic Server*.

## 2.5.7 WLST

This section describes new WebLogic Server WLST commands and changes to existing WebLogic Server WLST commands in this release.

The following WLST commands have been added in this release:

- `mergeDiagnosticData`—merges into a single file the set of CSV data files that were previously generated by the `dumpDiagnosticData` command.
- `getOldNodeManagerHome`—retrieves the previous Node Manager home directory for Node Manager upgrade.
- `getNodeManagerUpgradeType`—retrieves the Node Manager upgrade type for Node Manager upgrade.
- `getNodeManagerUpgradeOverwriteDefault`—gets the value of the Node Manager upgrade overwrite default flag.

The following modifications were made to existing WLST commands in this release.

- The `returnmap` argument has been added to the `start` command. This argument specifies whether or not to return a map containing the return values.
- The `Server` option has been added to the following diagnostic commands:
  - `saveDiagnosticImageCaptureEntryFile`
  - `saveDiagnosticImageCaptureFile`
  - `exportDiagnosticDataFromServer`
- The `Target` option has been added to the following diagnostic commands and should be used in place of the `Server` option for these commands:
  - `listSystemResourceControls`
  - `enableSystemResource`
  - `disableSystemResource`
  - `createSystemResourceControl`
  - `destroySystemResourceControl`
  - `captureAndSaveDiagnosticImage`
- The `outputdir` option has been added to the `captureAndSaveDiagnosticImage` command.
- The `enabled` option has been added to the `createSystemResourceControl` command for enabling a system control when it is created.
- The `dumpDiagnosticData` command has been updated so that if WLDF detects a change in the set of data collected between successive sampling periods, a new output file is created. The result may be multiple output files. (You can subsequently merge these files using the new `mergeDiagnosticData` command.)
- The `OldNodeManagerHome`, `NodeManagerUpgradeType`, and `NodeManagerUpgradeOverwriteDefault` names have been added as values for the

*optionName* argument to the `setOption` command. These names are used for setting Node Manager values during domain upgrade.

## 2.5.8 Upgrade

In WebLogic Server 12.1.3, the Reconfiguration Wizard now supports automatic reconfiguration of Node Manager from a per host Node Manager configuration to a per domain Node Manager configuration. The Node Manager home directory can be located in the domain home directory or in a custom location. In addition, per domain Node Manager configurations are automatically updated by the domain reconfiguration process.

For more information, see "Determining Node Manager Upgrade Procedure" in *Upgrading Oracle WebLogic Server*.

## 2.5.9 New Configuration Features

This section describes the configuration changes that have been implemented in WebLogic Server 12.1.3.

- [-Dcom.oracle.cie.config.showProfile Java System Property](#)
- [unpack Command -initialize Parameter](#)

### 2.5.9.1 -Dcom.oracle.cie.config.showProfile Java System Property

In WebLogic Server 12.1.3, the `-Dcom.oracle.cie.config.showProfile` Java system property has been added. When this property is set to `true` in the `CONFIG_JVM_ARGS` environment variable, the initial FMW Configuration Wizard screen contains options that allow you to specify whether the new domain is an expanded or compact domain.

For more information, see "Setting the `CONFIG_JVM_ARGS` Environment Variable" in *Creating WebLogic Domains Using the Configuration Wizard* and "Domain Topology Profiles" in *Domain Template Reference*.

### 2.5.9.2 unpack Command -initialize Parameter

An `initialize` parameter has been added to the `unpack` command. This parameter applies only to WebLogic domains in which other Fusion Middleware products are installed with WebLogic Server. By using this parameter during an `unpack` operation, you can initialize all or some of the `config-mapping.xml` values in the Service Table and OWSM data store, or not initialize any of these values.

For more information, see "Parameters of the `unpack` Command" in *Creating Templates and Domains Using the Pack and Unpack Commands*.

## 2.5.10 Deployment Validation Plug-In Support

WebLogic Server 12.1.3 includes an API for implementing a deployment validation plug-in that can intercept applications at the start of the deployment process and validate them prior to their deployment. The deployment validation plug-in support includes MBeans for enabling and configuring the plug-in.

For more information, see "Using a Deployment Validation Plug-In with WebLogic Server" in *Deploying Applications with the WebLogic Deployment API*.

## 2.6 Documentation Library Updates

The following table summarizes updates made to the WebLogic Server 12.1.3 documentation library since its initial release in June 2014:

Date	Description of Updates
April 2015	Added <i>Running Oracle WebLogic Server on Docker</i> to the documentation library.
May 2015	<ul style="list-style-type: none"><li>Added a list of videos to the Get Started page that show key WebLogic Server features and explain how to use them. See Featured Screenscasts at <a href="http://docs.oracle.com/middleware/1213/wls/index.html">http://docs.oracle.com/middleware/1213/wls/index.html</a>.</li><li>Added a section to this document that explains how to get access to the fixed bug list. See <a href="#">Section 2.8, "Obtaining Information about Patch Set Updates"</a>.</li><li>Added the following sections to <i>Release Notes for Oracle WebLogic Server</i>:<ul style="list-style-type: none"><li>"Availability of Fixes for IBM JDK 6.0 SR16 FP3 or JDK 7.0 SR8 FP10"</li><li>"Documentation Update for Specifying the Logging Implementation"</li><li>"Updates to the WebLogic Plug-In Enabled Attribute Description"</li></ul></li></ul>
June 2015	<ul style="list-style-type: none"><li>Added additional videos and created a new library page for WebLogic Server videos. See <a href="http://docs.oracle.com/middleware/1213/wls/wls-videos.htm">http://docs.oracle.com/middleware/1213/wls/wls-videos.htm</a>.</li><li>Added the following section to <i>Release Notes for Oracle WebLogic Server</i>:<ul style="list-style-type: none"><li>"weblogic.wsee.wstx.wsat.Transactional Annotation Incorrectly Marked as Deprecated"</li></ul></li><li>Updated the following sections to note that although JSSE supports Server Name Identification (SNI) in its SSL implementation, WebLogic Server does not support SNI:<ul style="list-style-type: none"><li>"Supported Standards" and "Using the JSSE-Based SSL Implementation" in <i>Administering Security for Oracle WebLogic Server</i></li><li>"JSSE and WebLogic Server" in <i>Developing Applications with the WebLogic Security Service</i></li><li>"Developing Clients That Use SSL" in <i>Developing Stand-alone Clients for Oracle WebLogic Server</i></li></ul></li></ul>

## 2.7 Standards Support, Supported Configurations, and WebLogic Server Compatibility

The following sections describe WebLogic Server standards support, supported system configuration, and WebLogic Server compatibility:

- [Standards Support](#)
- [Supported Configurations](#)
- [Licensing Information](#)
- [WebLogic Server Compatibility](#)
- [Database Interoperability](#)

### 2.7.1 Standards Support

WebLogic Server 12.1.3 supports the following standards and versions.

- [Java Standards](#)
- [Web Services Standards](#)

- Other Standards

### 2.7.1.1 Java Standards

Table 2–1 lists currently supported Java standards.

**Note:** See "WebLogic Server Security Standards" in *Administering Security for Oracle WebLogic Server* for the currently supported security standards, such as JAAS, JASPIC, JACC, JCE, and so forth.

While WebLogic Server 12.1.3 is a Java EE 6 compliant server, some Java EE 7 technologies have been added incrementally to this release; for example, JSR 356 Java API for WebSocket.

**Table 2–1 Java Standards Support**

Standard	Version
Contexts and Dependency Injection for Java EE	1.0
Dependency Injection for Java EE	1.0
Expression Language (EL)	2.2, 2.1, 2.0 Only JSP 2.0 and greater supports Expression Language 2.x.
Java API for JSON Processing (JSR-353)	1.0
Java API for XML-Based Web Services (JAX-WS)	2.2, 2.1, 2.0
Java API for RESTful Web Services (JAX-RS)	2.0
Java API for WebSocket	1.0
JavaBeans Activation Framework	1.1
Java EE	6.0
Java EE Application Deployment	1.2
Java EE Bean Validation	1.0
Java EE Common Annotations	1.1
Java EE Connector Architecture	1.6
Java EE EJB	3.1, 3.0, 2.1, 2.0, and 1.1
Java EE Enterprise Web Services	1.3, 1.2, 1.1
Java EE Interceptors	1.1
Java EE JDBC	4.0, 3.0
Java EE JMS	1.1, 1.0.2b
Java EE JNDI	1.2
Java EE JSF	2.1.*, 2.0, 1.2, 1.1
Java EE JSP	2.2, 2.1, 2.0, 1.2, and 1.1 JSP 1.2. and 1.1 include Expression Language (EL), but do not support EL 2.x or greater.
Java EE Managed Beans	1.0
Java EE Servlet	3.0, 2.5, 2.4, 2.3, and 2.2
Java RMI	1.0

**Table 2–1 (Cont.) Java Standards Support**

Standard	Version
JavaMail	1.4
JAX-B	2.2, 2.1, 2.0
JAX-P	1.3, 1.2, 1.1
JAX-R	1.0
JAX-RPC	1.1
JDKs	7, (6 and 7 clients)
JMX	2.0, when running WebLogic Server on JDK 6 or later 1.4, when running WebLogic Server on a JDK before JDK 6
JPA	2.1, 2.0., 1.0
JSR 77: Java EE Management	1.1
JSTL	1.2
Managed Beans	1.0
OTS/JTA	OTS 1.2 and JTA 1.1
RMI/IIOP	1.0
SOAP Attachments for Java (SAAJ)	1.3, 1.2
Streaming API for XML (StAX)	1.0
Web Services Metadata for the Java Platform	2.0, 1.1

### 2.7.1.2 Web Services Standards

For the current list of standards supported for WebLogic web services, see "Features and Standards Supported by WebLogic Web Services" in *Understanding WebLogic Web Services for Oracle WebLogic Server*.

### 2.7.1.3 Other Standards

Table 2–2 lists other standards that are supported in WebLogic Server 12.1.3.

---

**Note:** See "WebLogic Server Security Standards" in *Administering Security for Oracle WebLogic Server* for additional information on standards relating to security, such as SSL, TLS, and XACML, and so forth.

---

**Table 2–2 Other Standards**

Standard	Version
SSL	v3
X.509	v3
LDAP	v3
TLS	v1.1, v1.2
HTTP	1.1
SNMP	SNMPv1, SNMPv2, SNMPv3



**Table 2–2 (Cont.) Other Standards**

Standard	Version
xTensible Access Control Markup Language (XACML)	2.0
Partial implementation of Core and Hierarchical Role Based Access Control (RABC) Profile of XACML	2.0
Internet Protocol (IP)	Versions: <ul style="list-style-type: none"> <li>▪ v6</li> <li>▪ v4</li> </ul>

For more information about IPv6 support for all Fusion Middleware products, see the Oracle Fusion Middleware Supported System Configurations page on Oracle Technology Network.

## 2.7.2 Supported Configurations

For the most current information on supported configurations, see the Oracle Fusion Middleware Supported System Configurations page on Oracle Technology Network.

Oracle WebLogic Server 12.1.3 was certified on Java SE 7 when it was first made generally available. In addition, Oracle WebLogic Server 12.1.3 has been certified on Java SE 8 as follows:

- Oracle WebLogic Server 12.1.3 is certified on Java SE 8 Update 20 (u20) and will also be supported on later updates of Java SE 8.
- Oracle WebLogic Server 12.1.3 is certified on Java SE 8u20 on Windows, Linux, and Solaris with 64-bit JVMs.
- Oracle Coherence 12.1.3 is also certified on Java SE 8 u20. See the Oracle Coherence documentation for certification details and restrictions on application development.
- Other Oracle Fusion Middleware 12.1.3 products layered on Oracle WebLogic Server 12.1.3 are not supported on Java SE 8 unless specifically documented to be certified.
- See the Java SE 8 documentation for details about the features and capabilities available in Java SE 8. For information about garbage collection tuning options in Java SE 8, see the *HotSpot Virtual Machine Garbage Collection Tuning Guide*.
- Please note the following restrictions and advice when running WebLogic Server 12.1.3, and WebLogic Server 12.1.3 applications, on Java SE 8:
  - WebLogic Server 12.1.3 does not support applications using the new Java SE 8 fork/join and parallel streams features. Avoid these features when building WebLogic Server 12.1.3 applications using Java SE 8. The reason for this restriction is that the threads used by the fork/join thread pool will not be WebLogic Server managed threads. Any of the work performed in these threads may not be able to make use of WebLogic Server or Java EE facilities because the state of these threads, including security and transaction state, may not be created properly. Further, these threads will not be controlled by WebLogic Server Work Manager thread management facilities, possibly resulting in excessive thread usage.
  - When using Java SE 8, the use of "permgen" memory is no longer required. The Java command line `-XX:MaxPermSize` is ignored. Some of the WebLogic

Server shell scripts, such as `commEnv.sh`, will add it to the Java options but it will be ignored by the Java command.

- Customers should check third party vendor software that they are using for Java SE 8 compatibility. It may be necessary to upgrade to a later version of the software that correctly handles Java SE 8 classes and some software may not yet be compatible. For example, the current version of the open source tool "jarjar" does not work correctly with Java SE 8 yet.
- Java SE 8 has new APIs for JDBC 4.2 and is supported for WebLogic Server 12.1.3 running on Java SE 8 with a JDBC driver that supports JDBC 4.2. However, although the Oracle JDBC thin driver bundled with WebLogic Server is certified on Java SE 8, the Oracle JDBC thin driver does not support JDBC 4.2. The Derby 10.10 driver that is shipped with WebLogic Server 12.1.3 has been tested with JDBC 4.2. The corresponding Derby documentation is available at [http://db.apache.org/derby/docs/10.10/ref/rrefjdbc4\\_2summary.html](http://db.apache.org/derby/docs/10.10/ref/rrefjdbc4_2summary.html).
- WebLogic Server 12.1.3 can be installed using either Java SE 7 or Java SE 8. In order to migrate a domain using Java SE 7 and switch to Java SE 8, you should reset `JAVA_HOME` in any generated `DOMAIN/bin/setDomainEnv.sh` script.
- When using EJB 2.0 CMP, the compilation may fail with a JDT compilation error message. This problem is fixed with patch 18729264 available for WebLogic Server 12.1.3 from My Oracle Support.
- When running EJBCgen, the command may exit with a message indicating that it is not supported on JDK 8. When running `wlappc` on an EJB EAR file, it may fail with the following error: No EJBs found in the `ejb-jar` file. Both of these problems are fixed with patch 18428696 available for WebLogic Server 12.1.3 from My Oracle Support.
- When running using SSL connections with JCE on JDK 8, it may be necessary to install the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 8. You can download the JCE Unlimited Strength Jurisdiction Policy Files for JDK 8 at <http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html>.

### 2.7.3 Licensing Information

For the most current information on Oracle Fusion Middleware Licensing, see Licensing Information.

### 2.7.4 WebLogic Server Compatibility

For the most current information on compatibility between current version of WebLogic Server and previous releases, see "WebLogic Server Compatibility" in *Understanding Oracle WebLogic Server*.

### 2.7.5 Database Interoperability

The certification matrices and My Oracle Support Certifications define the following terms to differentiate between types of database support:

- [Application Data Access](#)
- [Database Dependent Features](#)

### 2.7.5.1 Application Data Access

Application Data Access refers to those applications that use the database for data access only and do not take advantage of WebLogic Server features that are Database dependant. WebLogic Server support of databases used for application data access only are less restrictive than for database dependent features.

WebLogic Server provides support for application data access to databases using JDBC drivers that meet the following requirements:

- The driver must be thread safe.
- The driver must implement standard JDBC transactional calls, such as `setAutoCommit()` and `setTransactionIsolation()`, when used in transactional aware environments.

Note the following restrictions:

- JDBC drivers that do not implement serializable or remote interfaces cannot pass objects to an RMI client application.
- Simultaneous use of automatic database connection failover and load balancing and global transactions (XA) with a highly-available (HA) DBMS architecture is supported with Oracle DB RAC only, and only for the Oracle DB RAC versions indicated on the **System** worksheet. These HA capabilities are only supported by Active GridLink for RAC and Multi Data Sources with RAC. These HA capabilities are not supported on other Oracle DB RAC versions or with other HA DBMS technologies on other non-Oracle DB products. Multi Data Sources are supported on other Oracle DB versions, and with non-Oracle DB technologies, but not with simultaneous use of automatic failover and load balancing and global transactions.
- Application data access to databases meeting the restrictions articulated above is supported on other Oracle DB versions, in addition to those documented in the certification matrix.
- WebLogic Type 4 JDBC drivers also support the following databases. For these databases, WebLogic Server supports application data access only, and does not support WebLogic Server database dependent features:
  - DB2 V9.1 for z/OS, DB2 V10 for z/O
  - Informix 11.0, Informix 11.5, Informix 11.7

### 2.7.5.2 Database Dependent Features

When WebLogic Server features use a database for internal data storage, database support is more restrictive than for application data access. The following WebLogic Server features require internal data storage:

- Container Managed Persistence (CMP)
- Rowsets
- JMS/JDBC Persistence and use of a WebLogic JDBC Store
- JDBC Session Persistence
- RDBMS Security Providers
- Database Leasing (for singleton services and server migration)
- JTA Logging Last Resource optimization
- JDBC TLog

## 2.8 Obtaining Information about Patch Set Updates

Information about bug fixes and patch set updates for WebLogic Server is available at My Oracle Support. Refer to the following articles:

Document ID	Title
1306505.1	Announcing Oracle WebLogic Server PSUs (Patch Set Updates)
2030245.1	Oracle WebLogic Server Patch Set Update 12.1.3.0.4 Fixed Bugs List
854428.1	Patch Set Updates (PSUs) for Oracle Products
950131.1	Error Correction Support Dates for Oracle WebLogic Server
1470197.1	Master Note on WebLogic Server Patch Set Updates (PSUs)

To learn more about known issues, refer to "Patch Set Update for WebLogic Server 12.1.3.0.4 Known Issues" in My Oracle Support note 2005626.1. To get access to these articles:

1. Log in to My Oracle Support at <https://support.oracle.com/>.  
If you are a new user, you must register at this site to gain access to these articles.
2. In the search field at the upper right, enter the support document ID you wish to read.

## 2.9 Deprecated Functionality (WebLogic Server 12c)

Information about deprecated functionality for WebLogic Server 12c Release 2 can be found on My Oracle Support at <https://support.oracle.com/>. Search for "Deprecated Features" in the **Search Knowledge Base** field.

### 2.9.1 WebLogic Full Client

The WebLogic full client, `wlfullclient.jar`, is deprecated as of WebLogic Server 12.1.3.

### 2.9.2 Log4j

The use of Log4j with the WebLogic logging service, as an alternative to Java logging, is deprecated as of WebLogic Server 12.1.3. Note that Log4j 2 and later is not supported in WebLogic Server.

### 2.9.3 LogMBean.ServerLoggingBridgeUserParentLoggersEnabled Attribute

The `ServerLoggingBridgeUserParentLoggersEnabled` attribute on the `LogMBean` is deprecated as of WebLogic Server 12.1.3.

### 2.9.4 weblogic.Admin

The `weblogic.Admin` utility is deprecated in WebLogic Server 12.1.3 and will be removed in a future release.

### 2.9.5 weblogic.management.username and weblogic.management.password

As of WebLogic Server 12.1.1, the boot username and password system properties `weblogic.management.username` and `weblogic.management.password` have been

deprecated and will be removed in a future release, and you will no longer be able to specify the username and password in the command for starting WebLogic Server in production mode.

As an alternative, Oracle recommends that you use the `boot.properties` file to specify the boot username and password for WebLogic Server. For more information about the `boot.properties` file, see "Boot Identity Files" in *Administering Server Startup and Shutdown for Oracle WebLogic Server*.

For information about other methods you can use to provide user credentials, see "Provide User Credentials to Start and Stop Servers" in *Administering Server Startup and Shutdown for Oracle WebLogic Server*.

## 2.9.6 Maven 11x Plug-In Deprecated

The `weblogic-maven-plugin` plug-in delivered in WebLogic Server 11g Release 1 is deprecated in this release. Oracle recommends that you instead use the WLS 12.1.2 Maven plug-in. See "Using the WebLogic Development Maven Plug-in" in *Developing Applications for Oracle WebLogic Server* for complete documentation.

## 2.9.7 JSP Tags for XML Processing

As of WebLogic Server 12.1.2, XSLT JSP tags and the WebLogic XSLT JSP Tag Library have been deprecated and will be removed in a future release. You can use JAXP to transform XML data. For more information, see "Transforming XML Documents" in *Developing XML Applications for Oracle WebLogic Server*.

## 2.9.8 `weblogic.websocket` and `weblogic.websocket.annotation`

As of WebLogic Server 12.1.3, the packages `weblogic.websocket` and `weblogic.websocket.annotation` are deprecated and will be removed in a future release. After these packages have been removed, you will no longer be able to use these packages for connections over the WebSocket protocol.

To ensure compatibility of your WebSocket applications with future releases of WebLogic Server, use the JSR 356 Java API for WebSocket instead of the deprecated packages.

For more information, including code examples, see "Using the WebSocket Protocol in WebLogic Server" in *Developing Applications for Oracle WebLogic Server*.

## 2.9.9 ActiveCache

ActiveCache is deprecated as of WebLogic Server 12.1.2.

## 2.9.10 Server Argument to WLST Diagnostics Commands

The `Server` argument to the following WLST diagnostics commands has been deprecated:

- `captureAndSaveDiagnosticImage`
- `createSystemResourceControl`
- `destroySystemResourceControl`
- `disableSystemResource`
- `enableSystemResource`

- `listSystemResourceControls`

The `Server` argument is being replaced by the `Target` argument. For more information, see "Diagnostics Commands" in *WLST Command Reference for WebLogic Server*.

## 2.9.11 URL Format for REST Management

The URL format for REST management APIs has changed in 12.1.3. The URL format introduced in 12.1.2 will continue to work, but is deprecated in WebLogic Server 12.1.3.

## 2.9.12 Node Manager Log File Rotation Properties

As of WebLogic Server 12.1.3, the following Node Manager log file rotation properties are deprecated and may be removed in a future release. For each deprecated property, the following table identifies the replacement property you should use instead.

Deprecated Property	Replacement Property
<code>FileSizeKB</code>	<code>process.FileSizeKB</code>
<code>FileTimeSpan</code>	<code>process.FileTimeSpan</code>
<code>FileTimeSpanFactor</code>	<code>process.FileTimeSpanFactor</code>
<code>NumberOfFilesLimited</code>	<code>process.NumberOfFilesLimited</code>
<code>RotatedFileCount</code>	<code>process.RotatedFileCount</code>
<code>RotatationTimeStart</code>	<code>process.RotationTimeStart</code>
<code>RotationType</code>	<code>process.RotationType</code>

For more information, see "Log File Rotation" in *Administering Node Manager for Oracle WebLogic Server*.

## 2.9.13 Node Manager Properties

As of WebLogic Server 12.1.3, the following Node Manager properties are deprecated and may be removed in a future release. For each deprecated property, the following table identifies the replacement property you should use instead.

Deprecated Property	Replacement Property
<code>CipherSuite</code>	<code>CipherSuites</code>
<code>CoherenceStartScriptEnabled</code>	<code>coherence.StartScriptEnabled</code>
<code>CoherenceStartScriptName</code>	<code>coherence.StartScriptName</code>
<code>DomainRegistrationEnabled</code>	To register multiple domains in the <code>nodemanager.domains</code> file, see "Configuring <code>nodemanager.domains</code> File" in <i>Administering Node Manager for Oracle WebLogic Server</i> .
<code>IfConfigDir</code>	<code>weblogic.IfConfigDir</code>
<code>Interface</code>	<code>InterfaceName=MinIP-MaxIP,NetMask=xxx</code>
<code>JavaHome</code>	Use <code>weblogic.startup.JavaHome</code> for WebLogic Server processes or <code>coherence.startup.JavaHome</code> for Coherence processes.

Deprecated Property	Replacement Property
NetMask	InterfaceName= <i>MinIP</i> - <i>MaxIP</i> , NetMask= <i>xxx</i>
StartScriptEnabled	weblogic.StartScriptEnabled
StartScriptName	weblogic.StartScriptName
StopScriptEnabled	weblogic.StopScriptEnabled
StopScriptName	weblogic.StopScriptName
UseMACBroadcast	weblogic.UseMACBroadcast

**Note:** WLST offline, as well as the pack and unpack commands, do not currently support the new Node Manager replacement properties introduced in WebLogic Server 12.1.3. When using these utilities, continue to use the preceding deprecated Node Manager properties, which remain fully supported in WebLogic Server 12.1.3.

For more information, see "Node Manager Properties" in *Administering Node Manager for Oracle WebLogic Server*.

## 2.10 Removed Functionality (WebLogic Server 12c)

The following functionality was removed from WebLogic Server in 12.1.3:

### 2.10.1 Specifying the Logging Implementation in the Administration Console

As of WebLogic Server 12.1.3, Log4j is deprecated. Related, the functionality to set the logging implementation using the WebLogic Server Administration Console was removed. As a result, the task describing this functionality in the *Oracle WebLogic Server Administration Console Online Help*, "Specify the logging implementation," is no longer applicable.

