Oracle® Fusion Middleware

What's New in Oracle WebLogic Server

14c (14.1.1.0.0)

F18284-23

January 2024

ORACLE®

Oracle Fusion Middleware What's New in Oracle WebLogic Server, 14c (14.1.1.0.0)

F18284-23

Contents

Preface

1   Introduction and Roadmap

2   What's New in Oracle WebLogic Server 14.1.1.0.0

Preface

This preface describes the document accessibility features and conventions used in this guide—*What's New in Oracle WebLogic Server 14.1.1.0.0*.

- Documentation Accessibility

- Diversity and Inclusion

- Conventions

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at `https://www.oracle.com/corporate/accessibility/`.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit `https://support.oracle.com/portal/` or visit `Oracle Accessibility Learning and Support` if you are hearing impaired.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.

Convention	Meaning
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1
Introduction and Roadmap

This chapter describes the contents and audience for this guide - *What's New in Oracle WebLogic Server 14.1.1.0.0*.
This chapter includes the following topics:

- Audience
- Guide to This Document

Audience

System administrators and operators responsible for monitoring and managing a WebLogic Server installation should use this document as a resource for understanding new features for all components of Oracle WebLogic Server 14c (14.1.1.0.0).

This document is relevant to all phases of a software project, from development through test and production phases.

Guide to This Document

This document is organized into two chapters.

- This chapter, Introduction and Roadmap, describes the scope and audience for this guide.
- What's New in Oracle WebLogic Server 14.1.1.0.0 describes the new and changed functionality in this WebLogic Server release.

2
What's New in Oracle WebLogic Server 14.1.1.0.0

Oracle WebLogic Server 14c (14.1.1.0.0) includes Java EE 8 full platform support, several open source tools for managing WebLogic Server in Kubernetes and Docker (see Manageability Improvements), and much more. This document describes the following new and changed functionality in Oracle WebLogic Server 14.1.1.0.0.

> **Note:**
>
> WebLogic Server 14.1.1.0.0 is a standalone WebLogic Server and Coherence only release. References to other Fusion Middleware (FMW) products do *not* indicate that they also are included in this release.

This chapter includes the following topics:

- WebLogic Server Update Summary
  The update summary lists the changes introduced in WebLogic Server 14.1.1.0.0 after the initial release.

- Java EE 8 Support
  Oracle WebLogic Server 14c (14.1.1.0.0) is a fully compatible implementation of the Java Platform, Enterprise Edition (Java EE) Version 8.0.

- JDK 11 Certification
  Oracle WebLogic Server 14c (14.1.1.0.0) is certified for use with JDK 11, in addition to JDK 8. Supported Oracle WebLogic Server 14c (14.1.1.0.0) clients are certified for use with JDK 11.0.6. A certified JDK is required for running the WebLogic Server installation program.

- WebLogic Server Slim Installer
  In Oracle WebLogic Server 14.1.1.0.0, in addition to the generic installer and the Fusion Middleware Infrastructure installer, you can use the slim installer to install and configure Oracle WebLogic Server and Coherence.

- Oracle GraalVM Enterprise Edition Certification
  Oracle WebLogic Server and Coherence 14.1.1.0.0 are certified to run on Oracle GraalVM Enterprise Edition.

- Runtime Improvements
  Oracle WebLogic Server 14c (14.1.1.0.0) builds on support from prior WebLogic Server versions to improve the reliability, availability, scalability, and performance of WebLogic Server applications with regard to the use of clustered environments, Oracle database features, and multi data center architectures.

- Manageability Improvements
  Oracle WebLogic Server 14c (14.1.1.0.0) continues to provide new management features that simplify the configuration, monitoring, and ongoing management of WebLogic Server domains and applications.

- Support for HTTP/2 and TLSv1.3 Protocols in Oracle WebLogic Server Proxy
  Plug-Ins
  The Oracle WebLogic Server 14.1.1.0.0 Proxy Plug-in is available for Apache
  HTTP Server on Linux and it supports the HTTP/2 and TLSv1.3 protocols.

- Documentation Update History
  The update history of the Oracle WebLogic Server documentation library
  summarizes the updates that have been made to various user and reference
  guides, as well as online help, for the initial release of version 14c (14.1.1.0.0).

- Standards Support, Supported Configurations, and WebLogic Server Compatibility
  Oracle WebLogic Server 14c (14.1.1.0.0) provides Java EE 8 full platform support,
  Java SE 11 certification, support for web services standards, support on multiple
  operating system and JVM platforms, and support for several security standards.

- WebLogic Server and Helidon Integration
  The integration capabilities between Oracle WebLogic Server and the Helidon
  microservices framework simplify application modernization with microservices, by
  allowing WebLogic-hosted applications to communicate and interoperate with
  Helidon-based microservices over different protocols.

- Deprecated Functionality (Oracle WebLogic Server 14c 14.1.1.0.0)
  Several functionalities and components have been deprecated in WebLogic
  Server 14c (14.1.1.0.0).

- Removed Functionality and Components
  Several components, deprecated in previous versions of WebLogic Server, are
  removed from Oracle WebLogic Server 14c (14.1.1.0.0).

WebLogic Server Update Summary

The update summary lists the changes introduced in WebLogic Server 14.1.1.0.0 after the initial release.

Feature	Description

Security	• The October 2023 Patch Set Update (PSU) includes the following changes:
	  – Adds support for SAML Single Logout (SLO) when WebLogic Server acts as a Service Provider. SAML SLO logs users out of all applications in their current SAML Single Sign-On session at once. See Configure SAML Single Logout in *Administering Security for Oracle WebLogic Server*.
	  – Adds support to configure SAML Single Sign-On using WLST offline. See Configuring SAML Single Sign On in *Understanding the WebLogic Scripting Tool*.
	• The April 2022 Patch Set Update (PSU) includes the following changes:
	  – Adds the KernelMBean attribute `RMIDeserializationMaxTimeLimit` and the `weblogic.rmi.stream.deserialization.timelimitmillis` system property which enable you to set a time limit when deserializing Java objects. See Setting the Deserialization Timeout Interval in *Administering Security for Oracle WebLogic Server*.
	• The October 2021 Patch Set Update (PSU) includes the following changes:
	  – Adds WebLogic Server Administration Console support for the JEP 290 allowlist model . See Using an Allowlist for JEP 290 Filtering in *Administering Security for Oracle WebLogic Server*.
	  – Enhances resolution guidance for security validation warnings in the Administration Console by providing links to relevant documentation. See Review Potential Security Issues in *Securing a Production Environment for Oracle WebLogic Server*
	• The July 2021 Patch Set Update (PSU) includes the following changes:
	  – Adds support for allowlists in JEP 290 filtering. When using the allowlist model, WebLogic Server and the customer define a list of the acceptable classes and packages that are allowed to be deserialized, and blocks all other classes. With the blocklist model, WebLogic Server defines a set of well-known classes and packages that are vulnerable and blocks them from being deserialized, and all other classes can be deserialized. While both approaches have benefits, the allowlist model is more secure because it only allows deserialization of classes known to be required by WebLogic Server and customer applications. See Using JEP 290 in Oracle WebLogic Server in *Administering Security for Oracle WebLogic Server*.
	  – Added support in WebLogic Server to detect the presence of a JEP 290 dynamic blocklist configuration file in the `ORACLE_HOME/oracle_common/common/jep290` directory, and block deserialization of classes and packages specified in the file. See Using a Dynamic Blocklist Configuration File in *Administering Security for Oracle WebLogic Server*.
	  – Added new security validation checks to determine if your domain meets Oracle recommended security guidelines. Warnings for failed validations are logged in the Administration Console. See Review Potential Security Issues in *Securing a Production Environment for Oracle WebLogic Server*.
	• The April 2021 Patch Set Update (PSU) includes the following changes:
	  – Support for dynamic blocklists, which provide the ability to update your JEP 290 blocklist filters by creating a configuration file that can be

Feature	Description
	updated or replaced while the server is running. See Using a Dynamic Blocklist Configuration File in *Administering Security for Oracle WebLogic Server*. – The ability to disable anonymous RMI T3 and IIOP requests. By default, WebLogic Server releases 14.1.1.0 and earlier enable clients to perform anonymous RMI requests. Disabling remote anonymous T3 and IIOP RMI requests will require that clients authenticate before invoking on WebLogic Server. Unauthenticated clients will be rejected. See Disable Remote Anonymous RMI T3 and IIOP Requests in *Securing a Production Environment for Oracle WebLogic Server*. – Support for RSA Crypto-J V6.2.5, RSA SSL-J V6.2.6, and RSA Cert-J V6.2.4.0.1. See Supported FIPS Standards and Cipher Suites in *Administering Security for Oracle WebLogic Server*. – Change to the default setting for the `ClasspathServletSecureModeEnabled` attribute in the `ServerTemplateMBean` from `false` to `true`. Setting the `ClasspathServletSecureModeEnabled` attribute to `true` enables secure mode by default and restricts access to several file types when using the `bea_wls_internal` web application. See Serving Resources from the CLASSPATH with the ClasspathServlet in *Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server*. • Completely reorganized the document *Securing a Production Environment for Oracle WebLogic Server* to more clearly highlight the steps required to lock down your WebLogic Server production environment. To ensure that your system is sufficiently protected, Oracle *strongly* recommends that all WebLogic Server customers review the contents of this document, specifically the topic Critical Tasks for Locking Down WebLogic Server. • Added documentation support for HTTP Strict Transport Security (HSTS), which is a web security policy mechanism that allows a web server to be configured so that web browsers, or other user agents, can access the server using only secure connections, such as HTTPS. See Using HTTP Strict Transport Security in *Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server*. • Added TLS v1.3 support for JDK 8 Update 261 (JDK 8u261) or later, and deprecated support for TLS v1.0 and v1.1. See Default Minimum TLS Protocol Version.
Certification on Oracle GraalVM Enterprise Edition	Oracle WebLogic Server and Coherence are now certified on Oracle GraalVM Enterprise Edition. See Oracle GraalVM Enterprise Edition Certification.
Network Channels for JTA Communication	Oracle WebLogic Server provides options to configure custom network channels for JTA communication. See Network Channels for JTA Communication.
Application Update Using Edition-Based Redefinition	Oracle WebLogic Server's Edition-Based Redefinition (EBR) feature enables you to perform an online update of the database component of an application to help minimize or eliminate application downtime. See Application Update Using Edition-Based Redefinition.
WebLogic Server slim installer	The slim installer is a lightweight installer that is much smaller than the generic or the Fusion Middleware Infrastructure installers. This installer does not have a graphical user interface and can be run from the command line only. See WebLogic Server Slim Installer.

Feature	Description
Support for HTTP/2 and TLSv1.3 protocols in WebLogic Server Proxy Plug-ins	The Oracle WebLogic Server 14.1.1.0.0 Proxy Plug-in is available for Apache HTTP Server on Linux and it supports the HTTP/2 and TLSv1.3 protocols. See Support for HTTP/2 and TLSv1.3 Protocols in Oracle WebLogic Server Proxy Plug-Ins.

Java EE 8 Support

Oracle WebLogic Server 14c (14.1.1.0.0) is a fully compatible implementation of the Java Platform, Enterprise Edition (Java EE) Version 8.0.

The key goals of the Java EE 8 platform are to modernize the infrastructure for enterprise Java for the cloud and microservices environments, emphasize HTML5 and HTTP/2 support, enhance ease of development through new Contexts and Dependency Injection features, and further enhance security and reliability of the platform. The Java EE 8 specification is available at `https://www.oracle.com/technetwork/java/javaee/overview/index-jsp-135147.html`.

Java EE 8 support provided in WebLogic Server 14c (14.1.1.0.0) is described in the following sections:

- Java API for JSON Binding 1.0 (JSR 367)
- Java API for JSON Processing 1.1 (JSR 374)
- Java API for RESTful Web Services 2.1 (JSR 370)
- JavaServer Faces 2.3 (JSR 372)
- Java Servlet 4.0 (JSR 369)
- Bean Validation 2.0 (JSR 380)
- Contexts and Dependency Injection 2.0 (JSR 365)
- Java EE Security API 1.0 (JSR 375)
- Sample Applications

Java API for JSON Binding 1.0 (JSR 367)

JSON-Binding (JSON-B) is a standard binding layer for converting Java objects to or from JSON messages.

Oracle WebLogic Server 14.1.1.0.0 supports the Java API for JSON Binding 1.0 (JSR 367) specification by including the JSR-367 reference implementation for use with applications deployed on a WebLogic Server instance.

JSON-B defines a default mapping algorithm for converting existing Java classes to JSON, while enabling developers to customize the mapping process through the use of Java annotations. For more information, see Java API for JSON Binding in *Developing Applications for Oracle WebLogic Server*.

Java API for JSON Processing 1.1 (JSR 374)

Oracle WebLogic Server 14.1.1.0.0 supports the JSON P 1.1 specification at `https://www.jcp.org/en/jsr/detail?id=374`. The JSON Processing 1.1 specification is based on the `javax.json` API, which supports new features such as JSON Pointer, JSON Patch, and JSON Merge Patch. These features are used to retrieve, transform, or manipulate values in an object model. For more information, see New Features for JSON Processing in *Developing Applications for Oracle WebLogic Server*.

Java API for RESTful Web Services 2.1 (JSR 370)

Oracle WebLogic Server 14.1.1.0.0 supports the Java API for RESTful Web Services (JAX-RS) 2.1 and Jersey 2.29 Reference Implementation (RI).

WebLogic Server supports server-sent events through the integration of the Jersey 2.29 RI:

- Server-sent events are used to push notifications asynchronously to the client over standard HTTP protocol.
- The server-sent events API is defined in the `javax.ws.rs.sse` package, which includes interfaces such as `Sse`, `SseEventSink`, `SseEvent`, `SseBroadcaster`, and `SseEventSource` for server-sent events.

For more information, see Using Server-Sent Events in *Developing and Securing RESTful Web Services for Oracle WebLogic Server* and the JSR 370 JAX-RS 2.1 specification.

JavaServer Faces 2.3 (JSR 372)

Oracle WebLogic Server 14.1.1.0.0 supports the JSF 2.3 specification at `https://jcp.org/en/jsr/detail?id=372`. The themes for this release are better integration with other APIs and leverage of Java SE 8 features. WebLogic Server support includes better CDI integration, better WebSocket integration, Ajax method invocation, and class-level bean validation. For more information, see JavaServer Faces (JSF) in *Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server*.

Java Servlet 4.0 (JSR 369)

Oracle WebLogic Server 14.1.1.0.0 supports the Servlet 4.0 specification (see `https://jcp.org/en/jsr/detail?id=369`), which introduces several new features, including support for HTTP/2, server push, HTTP trailer support, and mapping discovery. For more information, see What's New and Changed in Servlet 4.0 in *Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server*.

Bean Validation 2.0 (JSR 380)

Oracle WebLogic Server 14.1.1.0.0 supports Bean Validation 2.0 that defines a metadata model and API for validating data in JavaBeans components.

In 2.0, new data types and built-in constraints have been added for the purpose of validation. For more information, see `https://jcp.org/en/jsr/detail?id=380`.

Contexts and Dependency Injection 2.0 (JSR 365)

Oracle WebLogic Server 14.1.1.0.0 provides an implementation of the Contexts and Dependency Injection (CDI) for Java 2.0 specification.

The CDI specification for the Java EE platform defines a set of services for using injection to specify dependencies in an application. CDI uses the following specifications:

- JSR 330: Dependency Injection for Java

- Java EE 8 Managed Beans Specification, which is a part of the JSR 366: Java Platform, Enterprise Edition 8 (Java EE 8) Specification

- Interceptors specification, which is a part of JSR 345: Enterprise JavaBeans 3.2

CDI 2.0 provides the following features:

- The observer method to handle events. See Handling an Event in *Developing Applications for Oracle WebLogic Server*.

- The `fireAsync()` method to fire events asynchronously. See Sending an Event in *Developing Applications for Oracle WebLogic Server*.

- The standard API for bootstrapping a CDI container in Java SE. See Bootstrapping a CDI Container in *Developing Applications for Oracle WebLogic Server*.

Java EE Security API 1.0 (JSR 375)

Oracle WebLogic Server 14.1.1.0.0 supports the Java EE Security API 1.0 specification (see `https://www.jcp.org/en/jsr/detail?id=375`), which defines portable authentication mechanisms (such as `HttpAuthenticationMechanism` and `IdentityStore`), and an access point for programmatic security using the `SecurityContext` interface. You can use the built-in implementations of these APIs, or define custom implementations. For more information, see Using the Java EE Security API in *Developing Applications with the WebLogic Security Service*.

Sample Applications

The sample applications that can optionally be installed with WebLogic Server have been updated for Java EE 8, as described in the following sections:

- Avitek Medical Records — MedRec
- New Java EE 8 Examples

Avitek Medical Records — MedRec

Avitek Medical Records (or "MedRec") is a comprehensive educational sample application that demonstrates WebLogic Server and Java EE features, as well as best practices.

In Oracle WebLogic Server 14c (14.1.1.0.0), MedRec has been upgraded to demonstrate the following Java EE 8 features:

- JSF 2.3

- Injection of JSF Artifacts

- JSF Validation

- Bean Validation 2.0

- Servlet 4.0

- Other features, such as Optional, Stream API, and new APIs for date and time

For more information, see Avitek Medical Records.

New Java EE 8 Examples

New code examples have been added to show the following Java EE 8 features:

- CDI 2.0—Asynchronous events, observer ordering, and InterceptionFactory.

- JavaServer Faces 2.3—Direct support for WebSockets, class-level bean validation, the CDI-compatible `@ManagedProperty` annotation feature, and the new Java EE 8 date and time.

- Java Persistence 2.2—Injection in `@AttributeConverter` annotations, Java EE 8 date and time API, and retrieving the results of `Query` and `TypedQuery` as streams.

- Java EE Security API 1.0—Configuring a `DatabaseIdentityStore` to point at a back-end database and then using it as an `IdentityStore`.

- JAX-RS 2.1—New Server-Sent Events (SSE) and the Reactive Client API.

- JSON Binding 1.0—Using the Java API for JSON Binding (JSON-B) with JAX-RS.

- JSON Processing 1.1—Using JSON Patch, JSON Merge Patch, and JSON Pointer to update a JSON document.

- Servlet 4.0—Servlet Mapping API, HTTP/2 server push, and HTTP trailer headers.

For more information, see Java EE 8 Examples.

JDK 11 Certification

Oracle WebLogic Server 14c (14.1.1.0.0) is certified for use with JDK 11, in addition to JDK 8. Supported Oracle WebLogic Server 14c (14.1.1.0.0) clients are certified for use with JDK 11.0.6. A certified JDK is required for running the WebLogic Server installation program.

See the following topics:

- The Oracle Fusion Middleware Supported System Configurations page on Oracle Technology Network.

- The Java SE Downloads page on Oracle Technology Network from which Oracle Server JRE 11 is available, including release notes and installation instructions: http://www.oracle.com/technetwork/java/javase/downloads/index.html

  See also the Java Platform Group Product Management blog, Understanding the Server JRE.

- Roadmap for Verifying Your System Environment in *Installing and Configuring Oracle WebLogic Server and Coherence*

WebLogic Server Slim Installer

In Oracle WebLogic Server 14.1.1.0.0, in addition to the generic installer and the Fusion Middleware Infrastructure installer, you can use the slim installer to install and configure Oracle WebLogic Server and Coherence.

The slim installer does not contain examples, WebLogic Server Administration Console, WebLogic clients, Maven plug-ins and Java DB, and hence, has a smaller image size.

You can use this WebLogic Server installer for development, testing, and production purposes, in any infrastructure, such as, on premises (physical servers and virtual machines) or containers. Because it produces smaller WebLogic Server Docker or CRI-O images, this installer is particularly suitable for containers.

The slim installer file name is `fmw_14.1.1.0.0_wls_lite_quick_slim_generic.jar`.

See Obtaining the Oracle WebLogic Server and Coherence Distribution in *Installing and Configuring Oracle WebLogic Server and Coherence*.

Oracle GraalVM Enterprise Edition Certification

Oracle WebLogic Server and Coherence 14.1.1.0.0 are certified to run on Oracle GraalVM Enterprise Edition.

Oracle GraalVM Enterprise Edition is a high performance runtime platform built on Oracle's enterprise-class Java SE. Its optimizing compiler accelerates WebLogic applications by rearranging compiled code, aggressive method inlining, escape analysis, advanced vectorization and more. Based on internal testing, you should experience up to a 5-10% performance improvement.

For details, see *Running Oracle WebLogic Server and Coherence on GraalVM Enterprise Edition*.

Runtime Improvements

Oracle WebLogic Server 14c (14.1.1.0.0) builds on support from prior WebLogic Server versions to improve the reliability, availability, scalability, and performance of WebLogic Server applications with regard to the use of clustered environments, Oracle database features, and multi data center architectures.

These improvements are described in the following topics:

- Security
- RESTful Web Services
- Automatic Database Leasing Table Creation
- Network Channels for JTA Communication
- Application Update Using Edition-Based Redefinition

Security

The new security features provided in Oracle WebLogic Server 14c (14.1.1.0.0) are described in the following sections:

Default Minimum TLS Protocol Version

As of Oracle WebLogic Server 14.1.1.0.0, the default minimum version of the Transport Layer Security (TLS) protocol configured in WebLogic Server is Version 1.2. Oracle recommends the use of TLS v1.2 or later in a production environment. WebLogic Server logs a warning if the TLS version is set below 1.2.

> **Note:**
>
> - WebLogic Server supports TLS v1.3 with JDK 11, and JDK 8 Update 261 (JDK 8u261) or later. If you are running an earlier JDK version, then TLS v1.3 may not be available.
>
> - Support for TLS v1.0 and v1.1 is deprecated. Oracle strongly recommends that you do not use TLS v1.0 and v1.1. In addition, these versions may be disabled by default in certain JDK updates by the underlying JSSE provider.
>
> - WebLogic Server Web Server plug-ins currently support TLS v1.2 communication between Web Servers and WebLogic Server back ends. Customers who want to enable TLS v1.3 support through a load balancer to WebLogic Server back ends should evaluate load balancer alternatives such as hardware load balancers, or software load balancers such as NGINX.
>
> - When FIPS support is enabled, the RSA libraries support TLS v1.2.

See Specifying the SSL/TLS Protocol Version in *Administering Security for Oracle WebLogic Server*.

Default WebLogic Server Host Name Verifier

In Oracle WebLogic Server 14c (14.1.1.0.0), the default host name verifier has changed from the BEA host name verifier to a wildcard host name verifier.

See Using the Wildcard Host Name Verifier in *Administering Security for Oracle WebLogic Server*.

New RealmMBean Attributes for the Identity Assertion Cache Service

The following new configuration attributes were added to the `RealmMBean` to improve the performance of the cache services used by the Identity Assertion provider:

- `IdentityAssertionCacheEnabled`

- `IdentityAssertionCacheTTL`

- `IdentityAssertionDoNotCacheContextElements`

See Configuring Identity Assertion Performance in the Server Cache in *Administering Security for Oracle WebLogic Server*.

PKCS12 Default Keystore Type in JDK 11

PKCS12 is an extensible, standard, and widely-supported format for storing cryptographic keys. In JDK 11, the JDK default keystore type has changed from JKS to PKCS12.

The JDK default keystore type is determined by the default defined in the `keystore.type` property in the `java.security` file of your JDK installation. In JDK 8, the default is JKS. In JDK 11, the default is PKCS12. You can, however, explicitly specify the type of keystore you require. Existing keystores will not change.

For most WebLogic Server features that use a keystore, you can specify the keystore type using a command line option or a configuration setting. If you did not explicitly set the keystore type in your WebLogic Server configuration and you rely on the JDK default, when you upgrade to JDK 11 the JDK default keystore type may need to be updated. In this case, if you want to continue to use JKS as the keystore type, you can set the `storetype` property in the `java.security` file to JKS. If you prefer to use PKCS12, you can convert your JKS keystores using the `-importkeystore` option of the `keytool` utility. See the help for the `keytool` utility at https://docs.oracle.com/en/java/javase/11/tools/keytool.html.

Additionally, some features continue to use JKS as the default. For details, see Using the PKCS12 Keystore in WebLogic Server with JDK 11 in *Administering Security for Oracle WebLogic Server*.

New SSL System Passphrase Property for PKCS12 Trust Keystores

If you specify a trust keystore using the `-Dweblogic.security.SSL.trustedCAkeystore` command-line argument and the keystore type is PKCS12, then a password is required. You can specify the password using the `-Dweblogic.security.SSL.trustedCAkeystorePassPhrase` system property. The password is not necessary for JKS keystores. See SSL in *Command Reference for Oracle WebLogic Server*.

ImportPrivateKey Utility Uses JDK Default Keystore Type

In Oracle WebLogic Server 14.1.1.0.0 and later, the default value for the `storetype` argument of the `ImportPrivateKey` utility is determined by the default keystore type for the JDK.

The default keystore type for the JDK is defined by the `keystore.type` property in the `java.security` file. For JDK 8, the default is JKS. For JDK 11, the default is PKCS12. You can change the default by specifying the `storetype` property. See ImportPrivateKey in *Command Reference for Oracle WebLogic Server*.

Demo Certificates Contain the SAN Extension by Default

As a result of an enhancement in WebLogic Server 14.1.1.0.0, the demonstration certificates generated by the CertGen utility by default contain the fully-qualified DNS name in the Subject Alternative Name (SAN) extension value, along with the host name as the common name (CN) value. However, you can create your certificate without the SAN extension and disable the fully-qualified DNS name using the `-nosandnshost` option at the command line. Optionally, you can specify additional host names, or IP addresses, or both, in the SAN extension value using the `-a DNS:<hostname>,IP:<ip address>` option.

See Creating Demonstration Certificates Using CertGen in *Administering Security for Oracle WebLogic Server*.

Deprecated TLS Cipher Suites

Per Oracle security guidelines, the TLS cipher suites that are prefixed with `TLS_RSA_` or contain `_CBC_` are deprecated and are disabled by default. These disabled cipher suites are weak and do not provide sufficient security for your system. However, if necessary for your environment, you can enable these TLS cipher suites. See Deprecated Cipher Suites in *Administering Security for Oracle WebLogic Server*.

RESTful Web Services

Oracle WebLogic Server 14c (14.1.1.0.0) provides the following new and changed features for RESTful web services:

- Cross-Origin Resource Sharing (CORS) Support for WebLogic Server REST APIs. The Restful Management Service API includes new headers that are used specifically for CORS control. See Cross-Origin Resource Sharing for WebLogic Server REST APIs in *Administering Oracle WebLogic Server with RESTful Management Services*.

- Adds support for Java EE 8.

- Removes support for WebLogic Server Multitenant functionality and Resource Consumption Management.

Automatic Database Leasing Table Creation

Oracle WebLogic Server 14c (14.1.1.0.0) provides options to automatically create WebLogic cluster database leasing tables. See High Availability Database Leasing in *Administering Clusters for Oracle WebLogic Server* for the cluster configuration options related to database leasing.

Network Channels for JTA Communication

Oracle WebLogic Server 14c (14.1.1.0.0) provides options to configure custom network channels for JTA communication. The network channels are used for JTA interserver communication. See Configuring Network Channels for JTA Communication in *Developing JTA Applications for Oracle WebLogic Server*.

Application Update Using Edition-Based Redefinition

Edition-based redefinition (EBR) enables you to update the database component of an application while it is in use, thereby minimizing or eliminating application downtime.

When using EBR, a synchronized rolling update lets you update the configuration data concurrently with the new software version, thus ensuring that there is complete synchronization between the configuration and the software versions on the server.

For details, see Using Edition-Based Redefinition (EBR) to Update Applications in a Production Environment in *Deploying Applications to Oracle WebLogic Server*.

Manageability Improvements

Oracle WebLogic Server 14c (14.1.1.0.0) continues to provide new management features that simplify the configuration, monitoring, and ongoing management of WebLogic Server domains and applications.

These features are described in the following sections:

- Running Oracle WebLogic Server on Docker
- WebLogic Deploy Tooling
- WebLogic Image Tool
- WebLogic Server Kubernetes Operator
- WebLogic Monitoring Exporter
- WebLogic Logging Exporter

Running Oracle WebLogic Server on Docker

Docker is a Linux-based container technology that enables you to quickly create lightweight clustered and nonclustered WebLogic Server domain configurations on a single or multi host OS, or virtual machines, for either development or production environments. Oracle provides Dockerfiles and supporting scripts for building images of Oracle WebLogic Server. These images are built as an extension of existing Oracle Linux and Server JRE images. These scripts and Dockerfiles are available on GitHub at the following location: https://github.com/oracle/docker-images/tree/master/OracleWebLogic.

For information about using Docker with WebLogic Server, and the combinations of Oracle WebLogic Server, JDK, Linux and Docker versions that are certified for building your Docker images, see the Supported Virtualization and Partitioning Technologies for Oracle Fusion Middleware.

WebLogic Deploy Tooling

The Oracle WebLogic Deploy Tooling (WDT) simplifies the automation of WebLogic Server domain provisioning and applications deployment.

WDT creates a declarative, metadata model that describes the domain, applications, and resources used by applications. This metadata model makes it easy to provision, deploy, and perform domain lifecycle operations in a repeatable fashion. You can use WDT to migrate on-premises domain configuration and applications to a Docker image or a persistent volume in

Kubernetes. For complete documentation and samples, see the open source WebLogic Deploy Tooling project in GitHub.

WebLogic Image Tool

The Oracle WebLogic Image Tool lets you automate building, patching, and updating your WebLogic Server Docker images, including your own customized images.

With the WebLogic Image Tool, you can:

- Create a customized WebLogic Server and FMW Infrastructure Docker image.
- Patch a base install image of WebLogic Server or FMW Infrastructure.
- Patch and build a domain image of WebLogic Server or FMW Infrastructure using a WebLogic Deploy Tool (WDT) model.
- Deploy an application and update the domain configuration.

In addition, you can incorporate these use cases into an automated process for patching and updating your WebLogic Server infrastructure and applications running in Docker and Kubernetes. Find the open source WebLogic Image Tool GitHub project at `https://github.com/oracle/weblogic-image-tool`.

WebLogic Server Kubernetes Operator

The Oracle WebLogic Server Kubernetes Operator is an application-specific controller that extends Kubernetes to create, configure, and manage instances of complex applications. The operator follows the standard Kubernetes operator pattern, and simplifies the management and operation of WebLogic domains and deployments. The operator uses a common set of Kubernetes APIs to provide an improved user experience when automating operations such as provisioning, life cycle management, application versioning, product patching, scaling, and security.

The operator is developed as an open source project fully supported by Oracle. The fastest way to experience the operator is to follow the Quick Start guide. Alternatively, you can peruse the documentation, read the blogs, or try out the samples. For project scripts, additional samples, and source files, see the Oracle WebLogic Server Kubernetes Operator GitHub repository.

WebLogic Monitoring Exporter

The Oracle WebLogic Monitoring Exporter is a web application that you can deploy on a WebLogic Server instance that you want to monitor.

The exporter uses the WebLogic Server RESTful Management Interface for accessing runtime state and metrics and then exports Prometheus-compatible metrics, which can be displayed in Grafana dashboards for monitoring. For practical examples, see these blog posts: Exporting Metrics from WebLogic Server and Using Prometheus and Grafana to Monitor WebLogic Server on Kubernetes. For a detailed description of the Oracle WebLogic Monitoring Exporter, see the WebLogic Monitoring Exporter project in GitHub.

WebLogic Logging Exporter

The Oracle WebLogic Logging Exporter provides an easy to configure, robust, and production-ready solution to access WebLogic Server log information through Elasticsearch and displayed in Kibana dashboards.

Find the build and installation instructions in the open source WebLogic Logging Exporter project README at `https://github.com/oracle/weblogic-logging-exporter`.

Support for HTTP/2 and TLSv1.3 Protocols in Oracle WebLogic Server Proxy Plug-Ins

The Oracle WebLogic Server 14.1.1.0.0 Proxy Plug-in is available for Apache HTTP Server on Linux and it supports the HTTP/2 and TLSv1.3 protocols.

The Oracle WebLogic Server 14.1.1.0.0 Proxy Plug-in includes new features, such as server push functionality, support for TLSv1.3 cipher suites, support for certificates signed with RSASSA-PSS signature algorithm, and security improvements.

See Features of the 14.1.1.0.0 Plug-Ins in *Using Oracle WebLogic Server Proxy Plug-Ins*.

Documentation Update History

The update history of the Oracle WebLogic Server documentation library summarizes the updates that have been made to various user and reference guides, as well as online help, for the initial release of version 14c (14.1.1.0.0).

The following table summarizes updates made to the Oracle WebLogic Server 14.1.1.0.0 documentation library:

Date	Description of Updates
October 2023	Updated the following documents for changes introduced in the October 2023 Patch Set Update (PSU):
	• Added new topic, Configure SAML Single Logout, to *Administering Security for Oracle WebLogic Server* to describe new SAML Single Logout enhancement.
	• Added new topic, Configuring SAML Single Sign-On, to *Understanding the WebLogic Scripting Tool* to describe process to configure SAML Single Sign-On using WLST offline.
July 2023	Added a new guide *Integrating Oracle WebLogic Server with Helidon*.
October 2022	Added a new guide *Using Oracle WebLogic Server Proxy Plug-Ins*.
April 2022	Added a new topic Setting the Deserialization Timeout Interval in *Administering Security for Oracle WebLogic Server*.
November 2021	Added a new topic Warning About Using Local RDTs with Non-Durable MDBs in *Developing Message-Driven Beans for Oracle WebLogic Server*.

Date	Description of Updates
October 2021	Updated the following topics in *Administering Security for Oracle WebLogic Server* • October 2021 Patch Set Update (PSU) - Updated Using an Allowlist for JEP 290 Filtering to include support for using allowlists in the WebLogic Server Administration Console. • Updated the topic Configuring the RDBMS Security Store to clarify the procedure for creating a domain and using WLST offline to create the RDBMS security store. Updated the following topic in *Securing a Production Environment for Oracle WebLogic Server* • October 2021 Patch Set Update (PSU) - Updated Review Potential Security Issues to describe how to access more resolution information regarding security warnings in the WebLogic Server Administration Console.
July 2021	Updated the following documents for changes introduced in the July 2021 Patch Set Update (PSU): • *Administering Security for Oracle WebLogic Server* – Updated the topic Using JEP 290 in Oracle WebLogic Server to include support for allowlists and how to create them. – Updated the topic Securing the Domain After You Have Created It to include a reference to the new Security Warnings Report in the Administration Console. – Updated the topic Storing Private Keys, Digital Certificates, and Trusted Certificate Authority Certificates to include a link to the new certificate expiry notification topic. – Added a new topic Setting Certificate Expiry Notifications to describe how to set notifications regarding certificate expiry. – Updated the topic Replacing Expiring Certificates with a note regarding certificate expiry notification. • *Securing a Production Environment for Oracle WebLogic Server* – Updated the topic Use JEP 290 to Restrict Incoming Serialized Java Objects to include support for allowlists. – Added a new topic Review Potential Security Issues to describe the Security Warnings Report in the Administration Console. – Updated the topic Critical Tasks for Locking Down WebLogic Server to refer to the new Security Warnings Report in the Administration Console. – Updated the topic Understand How Domain Mode Affects the Default Security Configuration to refer to the new Security Warnings Report in the Administration Console.
April 2021	Updated the following documents for changes introduced in the April 2021 Patch Set Update (PSU): • *Administering Security for Oracle WebLogic Server* – Added a new topic to support dynamic blocklists UsIng a Dynamic Blocklist Configuration File. – Updated the topic Supported FIPS Standards and Cipher Suites to add support for RSA Crypto-J V6.2.5, RSA SSL-J V6.2.6, and RSA Cert-J V6.2.4.0.1. • Added a new topic Disable Remote Anonymous RMI T3 and IIOP Requests in *Securing a Production Environment for Oracle WebLogic Server* • Updated a note in the topic Serving Resources from the CLASSPATH with the ClasspathServlet in *Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server* for the change to the default setting for the `ClasspathServletSecureModeEnabled` attribute in the `ServerTemplateMBean`.

Date	Description of Updates
March 2021	Updated the following topics to add TLS v1.3 support in JDK 8, and deprecate support for TLS v1.0 and v1.1: • *Administering Security for Oracle WebLogic Server* – Supported Security Standards – Specifying the SSL/TLS Protocol Version • SSL in *Command Reference for Oracle WebLogic Server*
December 2020	• Added the following topics to support HTTP Strict Transport Security (HSTS): – Using HTTP Strict Transport Security in *Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server* – HTTP Strict Transport Security in *Command Reference for Oracle WebLogic Server* • Relocated the topic HTTP Proxy Servlet Parameters from *Using Oracle WebLogic Server Proxy Plug-Ins* to *Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server*. • Removed the following topics from *Upgrading Oracle WebLogic Server*: – Upgrading an 8.1 WebLogic Web Service to the WebLogic JAX-WS Stack – Upgrading a Domain that Uses an Evaluation Database • Removed the following topics about compatibility considerations for upgrading WebLogic Server from versions prior to 10.3.6, from *Upgrading Oracle WebLogic Server*: – Modifications to SSLMBean – New Web Services Features – Introduction of JSSE – Performance Enhancements for Security Policy Deployment – ActiveCache – Class Caching – Deprecated JDBC Drivers – Changes to weblogic.jms.extension API – Persistent Store Updates – Oracle Internet Directory and Oracle Virtual Directory Authentication Providers – CapacityIncrement Attribute – Middleware Home Directory – Resource Registration Name – Servlet Path Mapping – Evaluation Database Changed From PointBase to Derby
September 2020	• Reorganized the document *Securing a Production Environment for Oracle WebLogic Server*. • Relocated the topic An Important Note Regarding Null Cipher Use in SSL from *Securing a Production Environment for Oracle WebLogic Server* to *Administering Security for Oracle WebLogic Server*. • Added a new chapter, Using Edition-Based Redefinition (EBR) to Update Applications in a Production Environment, to *Deploying Applications to Oracle WebLogic Server*.
July 2020	Added a new topic, Configuring Network Channels for JTA Communication to *Developing JTA Applications for Oracle WebLogic Server*.
June, 2020	Added the following topics to support the Jython version upgrade in WebLogic Server 14.1.1.0.0: • Upgraded Version of Jython in *Upgrading Oracle WebLogic Server* • Behavior Changes in Jython version 2.7 in *Release Notes for Oracle WebLogic Server*

Date	Description of Updates
May, 2020	Added a new topic, Default Users, to *Securing Resources Using Roles and Policies for Oracle WebLogic Server*.
April, 2020	Added new guide *Running Oracle WebLogic Server and Coherence on GraalVM Enterprise Edition*.

Date	Description of Updates
March, 2020	Initial release. Library changes include:

- Due to the removal of WebLogic Server Multitenant functionality, as explained in WebLogic Server Multitenant Functionality and Resource Consumption Management, the following have been removed from the WebLogic Server 14.1.1.0.0 documentation:
  - *Using Oracle WebLogic Server Multitenant* has been removed from the library.
  - "Deploying Applications to Resource Groups and Templates" has been removed from *Deploying Applications to Oracle WebLogic Server*.
  - "Multitenancy Tuning Recommendations" has been removed from *Tuning Performance of Oracle WebLogic Server*.
  - "Partition Specific REST API Examples" has been removed from *Administering Oracle WebLogic Server with RESTful Management Services*.
  - "Using Proxy Data Sources" has been removed from *Administering JDBC Data Sources for Oracle WebLogic Server*.
- Prior to WebLogic Server 14.1.1.0.0, the Managed Server JMX notifications were enabled by default. Starting with WebLogic Server 14.1.1.0.0, the Managed Server JMX notifications are changed to be disabled by default. See Best Practices: Listening for WebLogic Server Events in *Developing Custom Management Utilities Using JMX for Oracle WebLogic Server*.
- SAML Credential Mapping provider Version 1 and SAML Identity Assertion provider Version 1 were deprecated in WebLogic Server 9.1. The following related topics have been removed from the *WebLogic Server Administration Console Online Help*:
  - "SAML Credential Mapping Provider V1: Common"
  - "SAML Credential Mapping Provider V1: Provider Specific"
  - "SAML Identity Asserter V1: Common"
  - "SAML Identity Asserter V1: Provider Specific"
- Due to the removal of the WebLogic Full client, IIOP-based clients, and the WebLogic JarBuilder Tool, as detailed in WebLogic Full and IIOP-Based Clients, the following chapters have been removed from *Developing Standalone Clients for Oracle WebLogic Server* in the WebLogic Server 14.1.1.0.0 documentation:
  - "Developing a WebLogic Full Client"
  - "Developing a Thin Client"
  - "WebLogic JMS Thin Client"
  - "Developing a Java SE Client"
  - "Developing a WLS-IIOP Client"
  - "Using the WebLogic JarBuilder Tool"
- Replaced the guide *Continuous Availability for Oracle WebLogic Server* with a new guide, *High Availability and Disaster Recovery Guide for Oracle WebLogic Server and Coherence*. This guide describes the high availability and disaster recovery features provided by Oracle WebLogic Server and Coherence 14c and how you can use these features, in combination with Oracle Database, to provide high availability and disaster recovery for three supported WebLogic Server and Coherence MAA architectures.
- Added the topic, "Understanding Cross-Domain Security," in *Developing JMS Applications for Oracle WebLogic Server*. Using a cross-domain security configuration, WebLogic Server establishes a security role for cross-domain users.
- Due to the removal of WebLogic jCOM, *Developing JCOM Applications for Oracle WebLogic Server* has been removed from the WebLogic Server 14.1.1.0.0 documentation library.

Date	Description of Updates
	• Active-Active XA Transaction Recovery (automated cross-site XA transaction recovery) has been removed from *Developing JTA Applications for Oracle WebLogic Server*.

Standards Support, Supported Configurations, and WebLogic Server Compatibility

Oracle WebLogic Server 14c (14.1.1.0.0) provides Java EE 8 full platform support, Java SE 11 certification, support for web services standards, support on multiple operating system and JVM platforms, and support for several security standards.

The following sections describe WebLogic Server standards support, supported system configuration, WebLogic Server compatibility, WebLogic Server Installation Support on Java SE 11 for AIX and zLinux, and WebLogic Server Installation Support on ARM-Based Oracle Cloud Infrastructure Ampere A1 (ARM OCI) Compute Instances:

- Standards Support
- Supported Configurations
- Licensing Information
- WebLogic Server Compatibility
- Database Interoperability
- WebLogic Server Installation Support on Java SE 11 for AIX and zLinux
- WebLogic Server Installation Support on ARM-Based Oracle Cloud Infrastructure Ampere A1 (ARM OCI) Compute Instances

Standards Support

WebLogic Server 14c (14.1.1.0.0) supports the following standards and versions:

- Java Standards
- Web Services Standards
- Other Standards
- Jython Version

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, the Java EE Security API, and so forth.

Table 2-1   Java Standards Support

Standard	Version
Java API for JSON Binding	1.0
Java EE Security API (JSR 375)	1.0
Batch Application Processing (JSR 352)	1.0
Contexts and Dependency Injection for Java EE	2.0, 1.1
Dependency Injection for Java EE	1.0
Concurrent Managed Objects (JSR 236)	1.0
Expression Language (EL)	3.0, 2.2, 2.1, 2.0 Only JSP 2.0 and greater supports Expression Language 2.x.
Java API for JSON Processing	1.1, 1.0
Java API for XML-Based Web Services (JAX-WS)	2.3, 2.2, 2.1, 2.0
Java API for RESTful Web Services (JAX-RS)	2.1, 2.0
Java API for WebSocket	1.1
JavaBeans Activation Framework	1.1
Java EE	8.0
Java EE Application Deployment	1.2
Java EE Bean Validation	2.0, 1.1
Java EE Common Annotations	1.3, 1.2
Java EE Connector Architecture	1.7
Java EE EJB	3.2, 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	1.2
Java EE JDBC	4.3
Java EE JMS	2.0, 1.1, 1.0.2b
Java EE JNDI	1.2
Java EE JSF	2.3, 2.2, 2.1.*, 2.0, 1.2, 1.1
Java EE JSP	2.3, 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	4.0, 3.1, 3.0, 2.5, 2.4, 2.3, and 2.2
Java RMI	1.0
JavaMail	1.6, 1.5
Java Transaction API	1.2
JAX-B	2.3, 2.2, 2.1, 2.0
JAX-P	1.3, 1.2, 1.1
JAX-R	1.0
JAX-RPC	1.1

Table 2-1   (Cont.) Java Standards Support

Standard	Version
JDKs	11.0 and 8.0 See JDK 11 Certification for details.
JMX	1.4
JPA	2.2, 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 1.2
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	2.1, 2.0, 1.1

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*.

Other Standards

Table 2-2 lists other standards that are supported in WebLogic Server 14c (14.1.1.0.0).

> **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
X.509	v3
LDAP	v3
TLS	v1.1, v1.2
HTTP	2.0, 1.1
SNMP	SNMPv1, SNMPv2, SNMPv3
xTensible Access Control Markup Language (XACML)	2.0

Table 2-2   (Cont.) Other Standards

Standard	Version
Partial implementation of Core and Hierarchical Role Based Access Control (RABC) Profile of XACML	2.0
Internet Protocol (IP)	Versions: • v6 • v4

Jython Version

WLST supports Jython. In Oracle WebLogic Server 14.1.1.0.0, the Jython version has been upgraded from version 2.2.1 to the current version 2.7.1. See:

- Upgraded Version of Jython in *Upgrading Oracle WebLogic Server* for information about the Jython version upgrade.

- Behavior Changes in Jython version 2.7 in *Release Notes for Oracle WebLogic Server* for issues caused by the Jython version upgrade and their workarounds.

Supported Configurations

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

Licensing Information

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

WebLogic Server Compatibility

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

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

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 for z/OS 10.1

  – Informix 11.7+

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

WebLogic Server Installation Support on Java SE 11 for AIX and zLinux

WebLogic Server 14c (14.1.1.0.0) is supported on AIX and zLinux systems on Java SE 8 and Java SE 11. See Oracle Fusion Middleware Supported System Configurations for more details.

You can use standard WebLogic Server 14c (14.1.1.0.0) generic installers to install WebLogic Server on AIX and zLinux systems on Java SE 8. Installation of WebLogic Server 14c (14.1.1.0.0) on AIX and zLinux systems on Java SE 11 requires a specific WebLogic Server installer and the following patch(es):

- The January 2022 PSU
- Patch 33466518

For development purposes, you can download and install the `fmw_14.1.1.0.0_wls_lite_generic_AIX_zLinux_JDK11.jar` file from the Oracle Fusion Middleware Software Downloads page at https://www.oracle.com/middleware/technologies/weblogic-server-downloads.html.

For production purposes, you can download the same installer from Oracle Software Delivery Cloud (OSDC) at https://edelivery.oracle.com/osdc/faces/Home.jspx.

For more details about the standard WebLogic Server installation procedures, see Installing and Configuring Oracle WebLogic Server and Coherence.

WebLogic Server Installation Support on ARM-Based Oracle Cloud Infrastructure Ampere A1 (ARM OCI) Compute Instances

WebLogic Server 14c (14.1.1.0.0) is supported on ARM OCI Compute instances. See Oracle Fusion Middleware Supported System Configurations for more details.

Installation of WebLogic Server 14c (14.1.1.0.0) on ARM OCI Compute instances requires specific WebLogic Server installers.

For development purposes, you can download and install the `fmw_14.1.1.0.0_wls_lite_generic_ARM_OCI.jar` or `fmw_14.1.1.0.0_wls_lite_quick_slim_generic_ARM_OCI.jar` file from the Oracle Fusion Middleware Software Downloads page at https://www.oracle.com/middleware/technologies/weblogic-server-downloads.html.

For production purposes, you can download the same installer from Oracle Software Delivery Cloud (OSDC) at https://edelivery.oracle.com/osdc/faces/Home.jspx.

For more details about the standard WebLogic Server installation procedures, see Installing and Configuring Oracle WebLogic Server and Coherence.

WebLogic Server and Helidon Integration

The integration capabilities between Oracle WebLogic Server and the Helidon microservices framework simplify application modernization with microservices, by allowing WebLogic-hosted applications to communicate and interoperate with Helidon-based microservices over different protocols.

With this integration, WebLogic and Helidon -based components can communicate and interoperate in the following ways:

- Bidirectional REST calls between WebLogic Server and Helidon.

- JMS message consumption and production by Helidon using WebLogic as the JMS provider.

- SOAP web service calls from Helidon to WebLogic Server Web Services.

- Single sign-on (SSO) between WebLogic Server and Helidon using Oracle Identity Cloud Service (IDCS).

- Distributed XA transaction coordination between WebLogic Server and Helidon using Oracle MicroTx Free.

For more information about this integration, see Integrating Oracle WebLogic Server with Helidon.

Deprecated Functionality (Oracle WebLogic Server 14c 14.1.1.0.0)

Several functionalities and components have been deprecated in WebLogic Server 14c (14.1.1.0.0).

- WebLogic Authorization and WebLogic Role Mapping Providers
- WebLogic HTTP Proxy Servlet
- ServletServlet
- WebLogic JMS Reconnect

WebLogic Authorization and WebLogic Role Mapping Providers

The WebLogic Authorization provider, which is referred to as the DefaultAuthorizer, and the WebLogic Role Mapping provider, which is referred to as the DefaultRoleMapper, have been deprecated in this release. Beginning with WebLogic Server 9.1, the XACML Authorization provider and the XACML Role Mapping provider are the default providers. For more information, see Configuring Authorization and Role Mapping Providers in *Administering Security for Oracle WebLogic Server*.

WebLogic HTTP Proxy Servlet

WebLogic `HttpProxyServlet` is deprecated.

Oracle recommends to use external load balancers such such HTTP load balancing functionality. Options include use of Oracle HTTP Server, Apache Web Server, hardware load balancers, OCI load balancer, or native Kubernetes load balancers when running in Kubernetes.

ServletServlet

`ServletServlet` is deprecated.

WebLogic JMS Reconnect

The WebLogic JMS Automatic Reconnect feature is deprecated. The JMS Connection Factory configuration, `javax.jms.extension.WLConnection` API, and `javax.jms.extension.JMSContext` API for this feature will be removed or ignored in a future release. Oracle recommends that client applications handle connection exceptions as described in Client Resiliency Best Practices in *Administering JMS Resources for Oracle WebLogic Server*.

Removed Functionality and Components

Several components, deprecated in previous versions of WebLogic Server, are removed from Oracle WebLogic Server 14c (14.1.1.0.0).

- OPatchAuto
- WebLogic Server Multitenant Functionality and Resource Consumption Management
- EJBGen
- WebLogic Full and IIOP-Based Clients
- WebLogic JMS Resource Adapter
- Oracle Traffic Director
- Compatibility Setting for JTA Security Interoperability Mode
- JMS Interop Modules
- Administration Console Extensibility
- WebLogic jCOM
- Automated Cross-Site XA Transaction Recovery
- DDInit and ddcreate Utilities

OPatchAuto

OPatchAutoFMW was removed in Oracle WebLogic Server 14c (14.1.1.0.0). Zero Downtime Patching continues to be supported, see the Zero Downtime Patching documentation.

WebLogic Server Multitenant Functionality and Resource Consumption Management

WebLogic Server Multitenant domain partitions, resource groups, resource group templates, virtual targets, resource override configuration MBeans, Resource Consumption Management, and proxy data sources have been removed from WebLogic Server as of version 14.1.1.0.0.

WebLogic Server Multitenant domain partitions enabled the configuration of a portion of a WebLogic domain that is dedicated to running application instances and related resources. Oracle recommends that customers that used domain partitions as a container dedicated to specific applications and resources consider the use of alternative container-based architectures, including the deployment of WebLogic applications and services in Docker

containers running in Kubernetes clusters. For more information, see Running Oracle WebLogic Server on Docker and WebLogic Server Kubernetes Operator .

EJBGen

EJBGen, an Enterprise JavaBeans 2.*x* code generator utility, has been removed from WebLogic Server as of version 14.1.1.0.0.

WebLogic Full and IIOP-Based Clients

The following WebLogic clients have been removed from WebLogic Server as of version 14.1.1.0.0:

- The WebLogic Full client (`wlfullclient.jar`) and its associated WebLogic JarBuilder Tool (`wljarbuilder.jar`).

- IIOP-based thin clients, including `wlclient.jar`, and the following clients that depend on it:

    - The JMS client, `wljmsclient.jar`

    - The JMS SAF client, `wlsafclient.jar`

    - The JMX client, `wljmxclient.jar`

- IIOP-based Java SE JDK clients (Java clients that use IIOP without any WebLogic JARs in their class path).

Oracle recommends using the thin T3 client (`wlthint3client.jar` with T3 protocol) or the install client (`weblogic.jar` with T3 or IIOP protocol) as a replacement for `wlfullclient.jar`, `wlclient.jar`, or the Java SE client (no WebLogic JAR) IIOP functionality. See Clients and Features in *Developing Standalone Clients for Oracle WebLogic Server*.

WebLogic JMS Resource Adapter

The WebLogic JMS resource adapter has been removed as of Oracle WebLogic Server 14.1.1.0.0.

Oracle recommends that you use either the thin T3 client or a message bridge to integrate applications running on non-WebLogic application servers through JMS. See the following topics:

- Developing a WebLogic Thin T3 Client in *Developing Standalone Clients for Oracle WebLogic Server*.

- Understanding the Messaging Bridge in *Administering the WebLogic Messaging Bridge for Oracle WebLogic Server*.

> ✎ **Note:**
>
> The WebLogic JMS resource adapter was supported only on Oracle GlassFish Server, so only users of GlassFish Server are affected.

Oracle Traffic Director

Oracle Traffic Director (OTD) has been removed from WebLogic Server as of version 14.1.1.0.0.

Oracle recommends using either one of the following for equivalent functionality: Oracle HTTP Server (good option for routing HTTP based traffic), Oracle Cloud Infrastructure (good option for routing both HTTP and TCP based traffic), Microsoft IIS Web Server, Apache HTTP Server Plug-In, or Traefik Kubernetes.

Compatibility Setting for JTA Security Interoperability Mode

The compatibility setting for JTA `Security Interoperability Mode` has been removed from WebLogic Server as of version 14.1.1.0.0.

JMS Interop Modules

The JMS Interop Modules have been removed from WebLogic Server as of version 14.1.1.0.0. If you have a module named `interop-jms.xml` in your `config.xml`, convert it to a regular system module. See JMS System Module Configuration.

Administration Console Extensibility

Console extensibility is removed as of WebLogic Server 14.1.1.0.0.

WebLogic jCOM

WebLogic jCOM is removed as of WebLogic Server 14.1.1.0.0.

WebLogic jCOM was provided as a migration path for interim solutions that required Java-to-COM integration. Oracle believes that web services and REST are the preferred way to communicate with Microsoft applications. Oracle recommends that you migrate legacy COM applications to .NET in order to use this type of communication.

Automated Cross-Site XA Transaction Recovery

Active-Active XA Transaction Recovery (automated cross-site XA transaction recovery) has been removed from WebLogic Server as of version 14.1.1.0.0. For more information on XA transaction recovery solutions, see Understanding XA Transaction Recovery in Disaster Recovery.

DDInit and ddcreate Utilities

DDInit and ddcreate, utilities for generating deployment descriptors for applications, have been removed from WebLogic Server as of version 14.1.1.0.0.