# Eclipse GlassFish Server Release Notes, Release 5.1

# Table of Contents


Eclipse GlassFish Server . . . 1
Preface . . . 2
- GlassFish Server Documentation Set . . . 2
- Related Documentation . . . 4
- Typographic Conventions . . . 5
- Symbol Conventions . . . 5
- Default Paths and File Names . . . 6

1 Eclipse GlassFish Server 6.1 Release Notes . . . 8
- Revision History . . . 8
- What's New in the GlassFish Server 5.1 Release? . . . 9
- Hardware and Software Requirements . . . 10
- Known Issues in GlassFish Server 5.1 . . . 15
- Restrictions and Deprecated Functionality . . . 17
- Documentation Errata . . . 20
- Features Available Only in the Full Platform . . . 20
- Java EE Standards Support . . . 21
- Java EE SDK . . . 23
- How to Report Problems and Provide Feedback . . . 24
- Additional Resources . . . 24

# Eclipse GlassFish Server

Release Notes

Release 5.1

Contributed 2018, 2019

These Release Notes provide late-breaking information about GlassFish Server 5.1 software and documentation. Also included are a summary of new product features in the 5.1 release, and descriptions and workarounds for known issues and limitations.

---

Eclipse GlassFish Server Release Notes, Release 5.1

Copyright © 2013, 2019 Oracle and/or its affiliates. All rights reserved.

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0, which is available at http://www.eclipse.org/legal/epl-2.0.

SPDX-License-Identifier: EPL-2.0

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.

# Preface

This documentation is part of the Java Enterprise Edition contribution to the Eclipse Foundation and is not intended for use in relation to Java Enterprise Edition or Orace GlassFish. The documentation is in the process of being revised to reflect the new Jakarta EE branding. Additional changes will be made as requirements and procedures evolve for Jakarta EE. Where applicable, references to Java EE or Java Enterprise Edition should be considered references to Jakarta EE.

Please see the Title page for additional license information.

This preface contains information about and conventions for the entire GlassFish Server Open Source Edition (GlassFish Server) documentation set.

GlassFish Server is developed through the GlassFish project open-source community at `https://javaee.github.io/glassfish/`. The GlassFish project provides a structured process for developing the GlassFish Server platform that makes the new features of the Java EE platform available faster, while maintaining the most important feature of Java EE: compatibility. It enables Java developers to access the GlassFish Server source code and to contribute to the development of the GlassFish Server. The GlassFish project is designed to encourage communication between Oracle engineers and the community.

The following topics are addressed here:


- GlassFish Server Documentation Set
- Related Documentation
- Typographic Conventions
- Symbol Conventions
- Default Paths and File Names


## GlassFish Server Documentation Set

The GlassFish Server documentation set describes deployment planning and system installation. For an introduction to GlassFish Server, refer to the books in the order in which they are listed in the following table.

| Book Title | Description |
|---|---|
| Release Notes | Provides late-breaking information about the software and the documentation and includes a comprehensive, table-based summary of the supported hardware, operating system, Java Development Kit (JDK), and database drivers. |

| Book Title | Description |
|---|---|
| Quick Start Guide | Explains how to get started with the GlassFish Server product. |
| Installation Guide | Explains how to install the software and its components. |
| Upgrade Guide | Explains how to upgrade to the latest version of GlassFish Server. This guide also describes differences between adjacent product releases and configuration options that can result in incompatibility with the product specifications. |
| Deployment Planning Guide | Explains how to build a production deployment of GlassFish Server that meets the requirements of your system and enterprise. |
| Administration Guide | Explains how to configure, monitor, and manage GlassFish Server subsystems and components from the command line by using the `asadmin` utility. Instructions for performing these tasks from the Administration Console are provided in the Administration Console online help. |
| Security Guide | Provides instructions for configuring and administering GlassFish Server security. |
| Application Deployment Guide | Explains how to assemble and deploy applications to the GlassFish Server and provides information about deployment descriptors. |
| Application Development Guide | Explains how to create and implement Java Platform, Enterprise Edition (Java EE platform) applications that are intended to run on the GlassFish Server. These applications follow the open Java standards model for Java EE components and application programmer interfaces (APIs). This guide provides information about developer tools, security, and debugging. |
| | |
| Embedded Server Guide | Explains how to run applications in embedded GlassFish Server and to develop applications in which GlassFish Server is embedded. |
| High Availability Administration Guide | Explains how to configure GlassFish Server to provide higher availability and scalability through failover and load balancing. |
| Performance Tuning Guide | Explains how to optimize the performance of GlassFish Server. |
| Troubleshooting Guide | Describes common problems that you might encounter when using GlassFish Server and explains how to solve them. |
| Error Message Reference | Describes error messages that you might encounter when using GlassFish Server. |
| Reference Manual | Provides reference information in man page format for GlassFish Server administration commands, utility commands, and related concepts. |
| Message Queue Release Notes | Describes new features, compatibility issues, and existing bugs for Open Message Queue. |
| Message Queue Technical Overview | Provides an introduction to the technology, concepts, architecture, capabilities, and features of the Message Queue messaging service. |

| Book Title | Description |
|---|---|
| Message Queue Administration Guide | Explains how to set up and manage a Message Queue messaging system. |
| Message Queue Developer's Guide for JMX Clients | Describes the application programming interface in Message Queue for programmatically configuring and monitoring Message Queue resources in conformance with the Java Management Extensions (JMX). |
| Message Queue Developer's Guide for Java Clients | Provides information about concepts and procedures for developing Java messaging applications (Java clients) that work with GlassFish Server. |
| Message Queue Developer's Guide for C Clients | Provides programming and reference information for developers working with Message Queue who want to use the C language binding to the Message Queue messaging service to send, receive, and process Message Queue messages. |

# Related Documentation

The following tutorials explain how to develop Java EE applications:

- Your First Cup: An Introduction to the Java EE Platform (`https://javaee.github.io/firstcup/`). For beginning Java EE programmers, this short tutorial explains the entire process for developing a simple enterprise application. The sample application is a web application that consists of a component that is based on the Enterprise JavaBeans specification, a JAX-RS web service, and a JavaServer Faces component for the web front end.
- The Java EE 8 Tutorial (`https://javaee.github.io/tutorial/`). This comprehensive tutorial explains how to use Java EE 7 platform technologies and APIs to develop Java EE applications.

Javadoc tool reference documentation for packages that are provided with GlassFish Server is available as follows.

- The API specification for version 8 of Java EE is located at `https://javaee.github.io/javaee-spec/javadocs/`.
- The API specification for GlassFish Server 5.0, including Java EE 8 platform packages and non-platform packages that are specific to the GlassFish Server product, is located at `https://javaee.github.io/javaee-spec/javadocs/`.

Additionally, the Java EE Specifications (`https://javaee.github.io/javaee-spec/Specifications`) might be useful.

For information about creating enterprise applications in the NetBeans Integrated Development Environment (IDE), see the NetBeans Documentation, Training & Support page (`http://www.netbeans.org/kb/`).

For information about the Apache Derby database for use with the GlassFish Server, see the Apache Derby product page (https://db.apache.org/derby/manuals/).

The Java EE Samples project is a collection of sample applications that demonstrate a broad range of Java EE technologies. The Java EE Samples are bundled with the Java EE Software Development Kit (SDK) and are also available from the Java EE Samples project page (https://github.com/javaee/glassfish-samples).

# Typographic Conventions

The following table describes the typographic changes that are used in this book.

| Typeface | Meaning | Example |
|---|---|---|
| `AaBbCc123` | The names of commands, files, and directories, and onscreen computer output | Edit your `.login` file.<br><br>Use `ls a` to list all files.<br><br>`machine_name% you have mail.` |
| `AaBbCc123` | What you type, contrasted with onscreen computer output | `machine_name% su`<br><br>`Password:` |
| AaBbCc123 | A placeholder to be replaced with a real name or value | The command to remove a file is `rm` filename. |
| AaBbCc123 | Book titles, new terms, and terms to be emphasized (note that some emphasized items appear bold online) | Read Chapter 6 in the User's Guide.<br><br>A cache is a copy that is stored locally.<br><br>Do not save the file. |

# Symbol Conventions

The following table explains symbols that might be used in this book.

| Symbol | Description | Example | Meaning |
|---|---|---|---|
| `[ ]` | Contains optional arguments and command options. | `ls [-l]` | The `-l` option is not required. |

| Symbol | Description | Example | Meaning |
|---|---|---|---|
| `{ \| }` | Contains a set of choices for a required command option. | `-d {y\|n}` | The `-d` option requires that you use either the `y` argument or the `n` argument. |
| `${ }` | Indicates a variable reference. | `${com.sun.javaRoot}` | References the value of the `com.sun.javaRoot` variable. |
| - | Joins simultaneous multiple keystrokes. | Control-A | Press the Control key while you press the A key. |
| + | Joins consecutive multiple keystrokes. | Ctrl+A+N | Press the Control key, release it, and then press the subsequent keys. |
| > | Indicates menu item selection in a graphical user interface. | File > New > Templates | From the File menu, choose New. From the New submenu, choose Templates. |

# Default Paths and File Names

The following table describes the default paths and file names that are used in this book.

| Placeholder | Description | Default Value |
|---|---|---|
| as-install | Represents the base installation directory for GlassFish Server. | Installations on the Oracle Solaris operating system, Linux operating system, and Mac OS operating system:<br><br>user's-home-directory`/glassfish6/glassfish`<br><br>Installations on the Windows operating system:<br><br>SystemDrive`:\glassfish6\glassfish` |
| as-install-parent | Represents the parent of the base installation directory for GlassFish Server. | Installations on the Oracle Solaris operating system, Linux operating system, and Mac operating system:<br><br>user's-home-directory`/glassfish6`<br><br>Installations on the Windows operating system:<br><br>SystemDrive`:\glassfish6` |
| domain-root-dir | Represents the directory in which a domain is created by default. | as-install`/domains/` |

| Placeholder | Description | Default Value |
|---|---|---|
| domain-dir | Represents the directory in which a domain's configuration is stored. | domain-root-dir`/`domain-name |
| instance-dir | Represents the directory for a server instance. | domain-dir`/`instance-name |

# 1 Eclipse GlassFish Server 6.1 Release Notes

GlassFish Server provides a lightweight, modular server for the development of Java Platform Enterprise Edition (Java EE) 8 applications. It is the Reference Implementation for Java EE. It delivers a highly productive platform for developing with the latest Java EE technologies.

For any issue or information on Eclipse GlassFish Server, see the GlassFish Server Community (`https://javaee.github.io/glassfish/`).

These Release Notes provide late-breaking information about GlassFish Server 5.1 software and documentation. These Release Notes include summaries of supported hardware, operating environments, and JDK and JDBC/RDBMS requirements. Also included are a summary of new product features in the 5.1 release, and descriptions and workarounds for known issues and limitations.

Refer to this document prior to installing, configuring, or using GlassFish Server 5.1 software. Consult this document periodically to view the most up-to-date product information.


- "Revision History"
- "What's New in the GlassFish Server 5.1 Release?"
- "Hardware and Software Requirements"
- "Known Issues in GlassFish Server 5.1"
- "Restrictions and Deprecated Functionality"
- "Documentation Errata"
- "Features Available Only in the Full Platform"
- "Java EE Standards Support"
- "Java EE SDK"
- "How to Report Problems and Provide Feedback"
- "Additional Resources"


## Revision History

This section lists the revision history for these Release Notes.

Table 1-1 Revision History

| Date | Description of Changes |
|---|---|
| September 2017 | Eclipse GlassFish Server 5.1. |

# What's New in the GlassFish Server 5.1 Release?

GlassFish is the Reference Implementation for Java EE. Java EE 8 introduces the JSON Binding API (JSON-B) for mapping between JSON text and Java objects, building on the JSON Processing API (JSON-P). Servlet has been enhanced with the addition of support for the new HTTP/2 protocol. JAX-RS adds support for server-sent events and, building on concurrency facilities added in Java SE 8, a reactive client API. The new Java EE Security API provides enhanced support for authentication and authorization in web modules, and also introduces APIs for access to identity stores. The Bean Validation facility is updated to reflect enhancements made in Java SE 8 and to extend the range of validated objects.

GlassFish Server 5.1 includes the following new and updated Java EE standards.

New Features

- Java EE Security API 1.0
- Java API for JSON Binding (JSON-B) 1.0

Updated

- Java API for JSON Processing (JSON-P) 1.1
- Java API for WebSocket 1.1
- Java API for RESTful Web Services (JAX-RS) 2.1
- Contexts and Dependency Injection for Java EE (CDI) 2.0
- Java Persistence API (JPA) 2.2
- Common annotations 1.3
- JavaServer Faces (JSF) 2.3
- Java Servlet 5.1
- Bean Validation (BV) 2.0
- Interceptors 1.2
- JavaMail 1.6

For a complete list of the Java EE technologies included in GlassFish Server 5.1, see Java EE Standards Support.

The main thrust of the Eclipse GlassFish Server 5.1 release is to provide an application server for developers to explore and begin exploiting the new and updated technologies in the Java EE 8 platform. Thus, the following features of GlassFish Server were not a focus of this release:

- Clusters and centralized management of standalone instances
- High availability features
- Upgrade
- Embedded Server

These features are included in the release, but they may not function properly with some of the new features added in support of the Java EE 8 platform.

# Hardware and Software Requirements

This section lists the requirements that must be met before installing Eclipse GlassFish Server Release 5.1 software.

The following topics are addressed here:


- Required JDK Versions
- Required Disk Space
- Required Free Ports
- Message Queue Broker Requirements
- Paths and Environment Settings for the JDK Software


## Required JDK Versions

Eclipse GlassFish Server Release 5.1 requires Oracle JDK 8 Update 144 or later.

Also be sure to see Paths and Environment Settings for the JDK Software for important JDK configuration instructions.

It is recommended that any machine that is hosting a GlassFish Server DAS or server instance have a minimum of 1 GB RAM.

## Required Disk Space

The download sizes for GlassFish Server 5.1 vary depending on the package you choose. The following are the approximate sizes of the ZIP packages for the Full and Web profiles:

- Full `*.zip`: 114 MB (136 MB unzipped)
- Web `*.zip`: 64.9 MB (82.9 MB unzipped)

The installation sizes will vary depending on your configuration, but the approximate amount of disk space used by GlassFish Server 5.1 is as follows:

- Full: 138 MB
- Web: 84.4 MB

## Required Free Ports

You must have sixteen unused ports available for the ports GlassFish Server uses. The installation program automatically detects ports that are in use and suggests currently unused ports for the default settings. The initial default port assignments are listed in the following table. If these default port numbers are in use, the installation program assigns a randomly selected port number from the dynamic port range. The selected port number might not be the next available port number.

Table 1-2 Default Port Assignments for GlassFish Server 5.1

| Port Number | Usage |
|---|---|
| 4848 | Administration Console |
| 8080 | HTTP |
| 8081 | HTTPS |
| 8686 | Pure JMX clients |
| 3700 | IIOP |
| 3820 | IIOP/SSL |
| 3920 | IIOP/SSL with mutual authentication |
| 22 | SSH port |
| 9009 | Java debugger |
| 7676 | JMS provider |
| Auto-generated from the operating system's dynamic port range | Message Queue TCP port |

| Port Number | Usage |
|---|---|
| Auto-generated from the operating system's dynamic port range | Message Queue Admin port |
| 9090 | GMS TCP start port |
| 9200 | GMS TCP end port |
| Auto-generated between GMS TCP start and end ports | GMS listener port |
| Auto generated between 2048 and 49151 | GMS multicast port |

In some situations, such as when multiple domains are running on a single host, port conflicts can arise in the auto-generated ports used by Message Queue and the GMS. To avoid these conflicts, you can configure the JMS host and the GMS to use specific ports.

#### To Configure Specific Ports for a JMS Host

When you create a JMS Host, GlassFish server automatically selects ports for the JMS provider (called the portmapper port in Message Queue terminology), the Message Queue TCP port and the Message Queue admin port.

To provide specific values for these ports, use the `--mqport` and `--property` options when creating the JMS host:

```
asadmin> create-jms-host --mqhost hostName --mqport portNumber \
--mquser adminUser --mqpassword adminPassword --target glassfishTarget \
--property imq\\.jms\\.tcp\\.port=tcpPort:imq\\.admin\\.tcp\\.port=adminPort \
jms-host-name
```

`--mqport portNumber`
: This option specifies the JMS provider port number.

`--property imq\\.jms\\.tcp\\.port=tcpPort:imq\\.admin\\.tcp\\.port=``adminPort`
: The `imq.jms.tcp.port` and `imq.admin.tcp.port` properties specify the TCP port and the admin port numbers. The double backslashes (`\\`) are used in the `--properties` option to escape the dots in the property names.

#### To Configure Specific GMS Ports for a Cluster

When you create a cluster, GlassFish server automatically selects a port for GMS multicast that does not conflict with the GMS multicast port of any other cluster in the domain. Additionally, when you start a cluster, the GMS automatically selects an available port in a specific range for its TCP listener.

If two or more domains are running on the same host, configure the clusters in the domains to ensure that no GMS port conflicts can arise among the clusters. To avoid possible port conflicts, use the `--multicast` and `--properties` options when creating the cluster:

```
asadmin> create-cluster --multicastport multicast-port \
--properties GMS_TCPSTARTPORT=start-port:GMS_TCPENDPORT=end-port \
cluster-name
```

`--multicastport multicast-port`

This option specifies the port number for the GMS to use for UDP multicast.

`--properties GMS_TCPSTARTPORT=start-port:GMS_TCPENDPORT=``end-port`

The `GMS_TCPSTARTPORT` and `GMS_TCPENDPORT` properties specify the range of port numbers the GMS is to use when selecting an available port for its TCP listener.

Note:

Though you can create a cluster, there is no support for configuration, as this has not been tested.

## Message Queue Broker Requirements

GlassFish Server 5.1 is now bundled with Message Queue (MQ) Broker 5.1.1. Refer to the `Open Message Queue Release Notes` for complete information about MQ Broker requirements.

## Paths and Environment Settings for the JDK Software

Ensure that your JDK configuration settings on all local and remote GlassFish Server hosts adhere to the guidelines listed below. Failure to adhere to these guidelines can cause various problems that may be difficult to trace.

The following topics are addressed here:

- Use the JDK Binaries
- Set the `JAVA_HOME` Environment Variable
- Set Other Environment Variables As Necessary

### Use the JDK Binaries

The following binary files that are used with GlassFish Server must come from the JDK software, not the Java Runtime Environment (JRE) software:

- `java`
- `keytool`

To meet this requirement, ensure that the `bin` directory for the JDK software precedes the `bin` directory for the JRE software in your path.

### Set the `JAVA_HOME` Environment Variable

Before performing any GlassFish Server installation or configuration procedures, set the `JAVA_HOME` environment variable on the GlassFish Server host machine to point to the correct Java version. Also be sure to add the `JAVA_HOME/bin` directory to the `PATH` variable for your environment. The `JAVA_HOME` variable must be set on all local and remote GlassFish Server hosts.

### Set Other Environment Variables As Necessary

All remote `asadmin` subcommands require the correct version of Java to be available on the affected remote machine. For example, when creating a cluster or server instance on a remote machine, the remote machine uses its local default Java installation, not the Java installation that is on the DAS. Errors will therefore occur if the remote machine uses the wrong Java version.

Depending on the remote subcommand, the errors may not occur when the subcommand is executed, but may occur later, when interacting with a configuration or resource created or modified by the subcommand. For example, when creating a clustered server instance on a remote machine, the error may only first appear when you attempt to deploy an application on that server instance.

This issue is more likely to be encountered when GlassFish Server is installed on the remote server by means of a ZIP file package as you do not have the option to specifically choose your Java version while unzipping a ZIP file.

Depending on what shell is invoked via SSH on the remote host, the `JAVA_HOME` and `PATH` environment variables may need to be explicitly set in `.bashrc`, `.cshrc`, or some other shell configuration file. This configuration file may differ from the one that is used when you log in to the machine, such as `.profile`.

Alternatively, you can specifically set the Java path with the `AS_JAVA` property in the in the as-install`/config/asenv.conf` file.

# Known Issues in GlassFish Server 5.1

This section describes known issues and any available workarounds for Eclipse GlassFish Server 5.1 software.

The following topics are addressed here:


- JAXB and JAX-WS no longer part of Java EE platform
- Resource validation property is enabled in the JVM option for deployment
- Update Tool and pkg Command no longer part of GlassFish Server
- Java DB has been replaced by Apache Derby


## JAXB and JAX-WS no longer part of Java EE platform

### Description

Jakarta XML Binding (previous JAXB) and Jakarta XML Web Services (previouly JAX-WS) are part of Java EE platform, but as optional technologies. However, the jars are packaged in GlassFish.

These jars are present only in the full profile of GlassFish and not part of web profile.

### Workaround

None

## Resource validation property is enabled in the JVM option for deployment

### Description

A new JVM option for deployment - deployment.resource.validation is introduced in GlassFish Server 5.1. This property is set to True by default so that each resource is validated during deployment time. This ensures that all resources are created beforehand. This property is applicable for administration server as well as instances when clusters are involved.

However, for deployment of applications containing embedded resource adapter, a connector resource is created after deployment. For the deployment of such applications to succeed, the server(s) must be started with this property set to false. For more information on JVM deployment options see `Administering JVM Options.`

#### Workaround

In case you do not want the resource validation to take place during the deployment, you can set this property value to False.

## Update Tool and pkg Command no longer part of GlassFish Server

#### Description

In previous releases, you could update your GlassFish Server software using the pkg command and the Update tool. Since the recent release of GlassFish server does not require the use of these features, they have been removed from the GlassFish server installation.

#### Workaround

No workaround.

## Java DB has been replaced by Apache Derby

#### Description

In the previous releases, Java DB was used as the database for GlassFish servers. With the release of GlassFish Server 5.1, Apache Derby 10.13.1.1 has replaced Java DB as the database for GlassFish servers.

#### Workaround

No workaround.

# Restrictions and Deprecated Functionality

This section describes restrictions and deprecated functionality in Oracle GlassFish Server 5.1.

The following topics are addressed here:


- `asadmin` Subcommands
- Deprecated, Unsupported, and Obsolete Options
- Applications That Use Apache Derby
- No Support for Client VM on Windows AMD64
- Metro Reliable Messaging in `InOrder` Delivery Mode


## `asadmin` Subcommands

In GlassFish Server 5.1, it is recommended that utility options of the `asadmin` command precede the subcommand. Utility options are options that control the behavior of the `asadmin` utility, as distinguished from subcommand options. Use of the following options after the subcommand is deprecated.

- `--host`
- `--port`
- `--user`
- `--passwordfile`
- `--terse`
- `--secure`
- `--echo`
- `--interactive`

## Deprecated, Unsupported, and Obsolete Options

Options in Table 1-3 are deprecated or no longer supported, or are obsolete and are ignored.

Table 1-3 Deprecated, Unsupported, and Obsolete Options for `asadmin` and Subcommands

| Option | Affected Subcommands |
|---|---|
| `--acceptlang` | Unsupported for the `create-virtual-server` subcommand. |
| `--acls` | Unsupported for the `create-virtual-server` subcommand. |
| `--adminpassword` | Unsupported for all relevant subcommands. Use `--passwordfile` instead. |
| `--autoapplyenabled` | Obsolete for the `create-http-lb` subcommand. |
| `--autohadb` | Obsolete for the `create-cluster` subcommand. |
| `--autohadboverride` | Obsolete for the `start-cluster` subcommand and the `stop-cluster` subcommand |
| `--blockingenabled` | Unsupported for the `create-http-listener` subcommand. |
| `--configfile` | Unsupported for the `create-virtual-server` subcommand. |
| `--defaultobj` | Unsupported for the `create-virtual-server` subcommand. |
| `--defaultvs` | Deprecated for the `create-http-listener` subcommand. Use `--default-virtual-server` instead. |
| `--description` | Obsolete for the `restore-domain` subcommand. |
| `--devicesize` | Obsolete for the `create-cluster` subcommand. |
| `--haadminpassword` | Obsolete for the `create-cluster` subcommand. |
| `--haadminpasswordfile` | Obsolete for the `create-cluster` subcommand. |
| `--haagentport` | Obsolete for the `create-cluster` subcommand. |
| `--haproperty` | Obsolete for the `create-cluster` subcommand. |
| `--hosts` | Obsolete for the `create-cluster` subcommand. |
| `--ignoreDescriptorItem` | Replaced by the all lowercase option `--ignoredescriptoritem` in the `set-web-context-param` subcommand and the `set-web-env-entry` subcommand. |
| `--mime` | Unsupported for the `create-virtual-server` subcommand. |
| `--password` | Unsupported for all remote subcommands. Use `--passwordfile` instead. |
| `--path` | Unsupported for the `create-domain` subcommand. Use `--domaindir` instead. |
| `--portbase` | Obsolete only for the `create-cluster` subcommand. This option is still valid in other subcommands such as `create-domain`, `create-instance`, and `create-local-instance`. |
| `--resourcetype` | Unsupported for all relevant subcommands. Use `--restype` instead. |
| `--retrievefile` | Obsolete for the `export-http-lb-config` subcommand. |
| `--setenv` | Obsolete for the `start-instance` subcommand. |

| Option | Affected Subcommands |
| --- | --- |
| `--target` | Obsolete only for the following subcommands:<br>• `create-connector-connection-pool`<br>• `create-resource-adapter-config`<br>• `delete-connector-connection-pool`<br>• `delete-connector-security-map`<br>• `delete-jdbc-connection-pool`<br>• `delete-resource-ref`<br>Replaced by an operand in the `list-custom-resources` subcommand and the `list-jndi-entries` subcommand: |

## Applications That Use Apache Derby

The directory location of Apache Derby in GlassFish Server 5.1 has changed from its location in previous installations. Suppose that you have deployed applications that use Apache Derby databases in your previous server installation, and you upgrade your existing installation to GlassFish Server 5.1. If you run the `asadmin start-database` command and successfully start Apache Derby, you could run into problems while trying to run applications that were deployed on your previous server installation.

To solve this problem, you can copy the `databases` directory from your previous installation to as-install`/databases`. Make sure the database is not running when you do this.

Alternatively, you can perform these steps:

1. Use the `asadmin start-database` command with the `--dbhome` option pointing to the `databases` directory in the older version of Apache Derby. For example:

```
asadmin start-database --dbhome c:\glassfish\databases
```

1. After upgrade, start GlassFish Server 5.1.

## No Support for Client VM on Windows AMD64

By default, the GlassFish Server DAS uses the Client VM to achieve best startup and deployment performance. If you are using Windows AMD64, edit the `domain.xml` file to remove the line `<jvm-options>-client<jvm-options>`. In this case, JVM ergonomics chooses the appropriate kind of VM for the given platform. Note that server instances use the Server VM by default.

For more information about platform support, see Ergonomics in the 5.1 Java Virtual Machine.

### Metro Reliable Messaging in `InOrder` Delivery Mode

The Metro Reliable Messaging in `InOrder` Delivery mode has not been tested for high availability in GlassFish Server 5.1. The feature may work, but it has not been formally tested and is therefore not a supported feature.

### No Support for Kerberos on AIX

GlassFish Server 5.1 does not support Kerberos on the AIX platform.

For the complete report about this issue, see `Issue-16728`

## Documentation Errata

This section describes documentation errata.

- Upgrading to Oracle GlassFish Server Is Not Necessary

### Upgrading to Oracle GlassFish Server Is Not Necessary

The Eclipse GlassFish Server Administration Guide discusses upgrading Eclipse GlassFish Server to Oracle GlassFish Server. GlassFish Server 4.x is only an open source release, so this upgrade is not necessary.

Upgrading may not work for GlassFish Server 5.1

## Features Available Only in the Full Platform

The following features of GlassFish Server 5.1 are available only in the Full Platform:

- EJB features that make up the full EJB 3.2 API, such as remote EJB components, message-driven beans, web service EJB endpoints, and the EJB Timer Service
  The EJB 3.2 Lite specification is supported in the Web Profile. This specification allows enterprise beans within web applications and includes support for local stateless session beans, stateful

session beans, and singleton session beans.

- Application Client Container
- JMS resources
- Web services
  In the Web Profile, a servlet or EJB component cannot be a web service endpoint. The `sun-web.xml` and `sun-ejb-jar.xml` elements that are related to web services are ignored.
- Message security
- JavaMail resources

Connector modules that use only outbound communication features and work-management that does not involve inbound communication features are supported in the Web Profile. Other connector features are supported only in the Eclipse GlassFish Server 5.1 full platform.

# Java EE Standards Support

Table 1-4 lists the Java EE standards implemented in GlassFish Server 5.1. The table also indicates the distributions in which the implementation of a standard is available.

- X indicates that the implementation is available in the distribution.
- - indicates that the implementation is not available in the distribution.

Table 1-4 Java EE Standards Implementations in GlassFish Server 5.1

| Java EE Standard | Java Specification Request (JSR) | GlassFish Server 5.1 Full Platform | GlassFish Server 5.1 Web Profile |
|---|---|---|---|
| Java Platform, Enterprise Edition 8 (https://javaee.github.io/javaee-spec/javadocs/) | JSR 366 | X | X |
| Java EE Security API | JSR 375 | X | X |
| Batch Applications for the Java Platform 1.0 | JSR 352 | X | - |
| Concurrency Utilities for Java EE 1.0 | JSR 236 | X | - |
| Java API for JSON Processing 1.1 | JSR 374 | X | X |
| Java API for JSON Binding 1.0 | JSR 367 | X | X |
| Java API for WebSocket 1.1 | JSR 356 | X | X |
| https://javaee.github.io/servlet-spec/ (https://javaee.github.io/servlet-spec/)/ | JSR 369 | X | X |

| Java EE Standard | Java Specification Request (JSR) | GlassFish Server 5.1 Full Platform | GlassFish Server 5.1 Web Profile |
|---|---|---|---|
| JavaServer Pages 2.3 | JSR 245 | X | X |
| Expression Language 3.0 | JSR 341 | X | X |
| Debugging Support for Other Languages 1.0 | JSR 45 | X | X |
| Standard Tag Library for JavaServer Pages 1.2 | JSR 52 | X | X |
| JavaServer Faces 2.3(https://github.com/javaserverfaces/) | JSR 372 | X | X |
| Common Annotations for the Java Platform 1.2 | JSR 250 | X | X |
| Java Transaction API 1.2 (https://github.com/javaee/jta-spec) | JSR 907 | X | X |
| Java Persistence API 2.2 (https://github.com/javaee/jpa-spec) | JSR 338 | X | X |
| Managed Beans 1.0 | JSR 316 | X | X |
| Interceptors 1.2 | JSR 318 | X | X |
| Dependency Injection for Java 1.0 | JSR 330 | X | X |
| Enterprise JavaBeans 3.2 Lite (https://github.com/javaee/ejb-spec) | JSR 345 | X | X |
| Enterprise JavaBeans 3.2 Full API (https://github.com/javaee/ejb-spec) | JSR 345 | X | - |
| Contexts and Dependency Injection for Java EE 2.0 | JSR 365 | X | X |
| Java API for RESTful Web Service (JAX-RS) 2.1 | JSR 370 | X | X |
| Bean Validation 2.0 | JSR 380 | X | X |
| Java EE Connector Architecture 1.7 (https://github.com/javaee/connector-spec) | JSR 322 | X | X* |
| Java API for XML-Based Web Services (JAX-WS) 2.3(https://github.com/javaee/metro-jax-ws) | JSR 224 | X | - |
| Java Architecture for XML Binding (JAXB) 2.2 (https://github.com/javaee/jaxb-v2) | JSR 222 | X | - |
| Implementing Enterprise Web Services 1.4 | JSR 109 | X | - |
| Web Services Metadata for the Java Platform 2.1 | JSR 181 | X | - |
| https://javaee.github.io/jms-spec/ (https://javaee.github.io/jms-spec/) | JSR 343 | X | - |
| https://javaee.github.io/javamail/ (https://javaee.github.io/javamail/) | JSR 919 | X | - |
| https://javaee.github.io/jacc-spec/ (https://javaee.github.io/jacc-spec/) | JSR 115 | X | - |

| Java EE Standard | Java Specification Request (JSR) | GlassFish Server 5.1 Full Platform | GlassFish Server 5.1 Web Profile |
|---|---|---|---|
| Java Authentication Service Provider Interface for Containers 1.1 | JSR 196 | X | - |
| Java EE Application Deployment 1.2 | JSR 88 | X | - |
| J2EE Management 1.1 | JSR 77 | X | - |
| Java API for XML-Based Remote Procedure Calls (JAX-RPC) 1.1(https://github.com/javaee/jax-rpc-ri/) | JSR 101 | X | - |
| Java API for XML-Based Registries (JAXR) 1.0 | JSR 93 | X | - |

* Standalone Connector 1.7 Container only.

Building on these standards, GlassFish Server 5.1 provides a number of extensions, including the following:

- Ajax (asynchronous JavaScript and XML): Retrieves and displays new data for a portion of a web page without affecting the rest of the page.
- Metro: A web services stack that implements Java Architecture for XML Binding (JAXB) and Java APIs for XML Web Services 2.3 (JAX-WS 2.3).
- Grizzly: A framework for building scalable and robust servers using New I/O (NIO) APIs, which make scaling to thousands of users possible. The ability to embed components that support HTTP, Bayeux Protocol, Java Servlet API, and Comet is provided.

# Java EE SDK

GlassFish Server 5.1 is available as part of the Java EE 8 SDK. The following versions of the Java EE 8 SDK are available:

- Java EE 8 SDK: This version includes GlassFish Server 5.1. This version is designed for developers who require the full set of Java EE APIs for enterprise application development.
- Java EE 8 Web Profile SDK: This version includes GlassFish Server 5.1 Web Profile. This version contains web technologies that are subset of the Full Platform and is designed for developers who do not require the full set of Java EE APIs.

More information about the Java EE SDK distributions are available at Java EE Reference at a Glance (http://www.oracle.com/technetwork/java/javaee/documentation/index.html). Java EE SDK distributions are available from the Java EE SDK Downloads page (http://www.oracle.com/technetwork/java/javaee/downloads/index.html).

# How to Report Problems and Provide Feedback

If you have problems with GlassFish Server 5.1, provide feedback through one of the following mechanisms:

- GlassFish Server forum (https://javaee.groups.io/g/glassfish) — A variety of GlassFish Server community for various interests and feedback
- GlassFish Issue tracker (https://github.com/javaee/glassfish/issues) — GlassFish Server project dashboards and issue tracking database

# Additional Resources

Useful information can be found at the following locations:

- GlassFish Server Community (https://javaee.github.io/glassfish/)
- Oracle Developer Information (http://www.oracle.com/technetwork/index.html)
- Oracle product documentation (http://www.oracle.com/technetwork/indexes/documentation/index.html)